AMSTRAD CPC

AMSTRAD CENT POUR CENT

HITS
PREHISTORIK 2
CRAZY CARS 3
SUPER CAULDRON

INFERNAL
FAITES UN CPC+
AVEC UNE GX-4000

LISTING
UN VRAI
GESTIONNAIRE
DE SPRITES

3615
LES BEST OF DU CPC

POKES
LA SUITE DES
COLLECTORS

N° 48 25 F
TRIMESTRIEL
JUILLET 93

ISSN 0988 8160 - 48 - 25 F - 180 FL - 183 FB - 1660 CFA



SOMMAIRE

EDITORIAL

Tout s'écroule ! L'économie patine. L'Europe déprime. Les milices s'étripent des Balkans à la Somalie... Et pourtant, il existe des hommes heureux... les amstradiens qui vont oublier leurs tracas à la plage, grâce à leur machine de rêve et leur magazine préféré. Oui, les vacances sont là, et *Amstrad CPC* n'a pas lésiné sur les moyens de vous distraire. Jugez plutôt.

Vous avez été très nombreux à nous demander des renseignements concernant le tour de magie qui permet de transformer une GX-4000 associée à un CPC classique en un véritable CPC+. Réjouissez-vous, sortez la caisse à outils et foncez à la rubrique « fer à souder » où toutes les manipulations pour cette opération délicate sont détaillées.

Comme d'habitude, vos testeurs préférés ont déniché les jeux à ne surtout pas manquer. Et croyez-nous, leur collecte fut abondante.

Grâce à notre super listing, et à nos rubriques d'initiation au Basic, à l'assembleur ou au CPC+, vous pourrez enfin créer tous les jeux dont vous rêvez et jouer comme vous aimez.

Les petits malins retrouveront avec plaisir nos « petits prix » qui leur permettront de s'éclater tout en bronzant pour trois fois rien.

En attendant le prochain numéro, prélassez-vous et n'oubliez le 3615 ACPC, le rendez-vous des mordus 24 heures sur 24.

Bonnes vacances, nous nous reverrons au mois d'octobre pour une rentrée pleine de surprises.

Amstrad Cent Pour Cent

AMSTRAD CENT POUR CENT est une publication de **MÉDIA SYSTÈME ÉDITION** 19, rue Louis-Pasteur, 92513 BOULOGNE CEDEX - Tél. : 41 10 20 30.

Directeur de la publication et rédacteur en chef : Alain KAHN.

Secrétaire de rédaction : Ivan GAUCHER.

Ont participé à ce numéro : Denis JARRIL, Les LOGON SYSTEM, Christophe JORGE, Xavier LAMBERT, LUDOTRONIC, Claude LESUEUR, MEGADETH, Christophe POTTIER, Alain MASSOUMIPOUR, Valérie RICHARD (traduction). **Illustrateur :** MYKAÏA. **Secrétariat :** Christine NABAIS. **Permanence téléphonique** tous les mercredis de 10 h à 18 h.

Publicité : tél. : 41 10 20 40. **Responsable de la publicité :** Perrine MICHAELIS. **Chef de publicité :** Nezhia BOUBEKRI.
Responsable marketing et télématique : Barbara RING-RÉMY. **Minitel :** 3615 ACPC.

Responsable de la fabrication : Isabelle DERVEAUX-BERTÉ. **Assistante de fabrication :** Valérie RICHARD. **Photogravure :** Pi-M. **Impression :** IEI Lisses.
Service abonnements : BP 4 - 60700 Sacy le Grand - Tél. : (16) 44 69 26 26.

Média Système Edition, SA au capital de 250 000 F RCS Nanterre B341 547 024. **Président-directeur général :** Alain KAHN.
Responsable financière et comptable : Sylvie BRUTINAUD. **Comptabilité :** Françoise LE METAYER. **Administration :** Jacqueline SÉVASTELLIS.
Commission paritaire en cours - Distribution NMPP - Dépôt légal janvier 1993.

Couverture : Prehistorik 2.

PRE-HISTORIK 2

Elmsoft, excellent démomaker, commence à se faire une véritable réputation en matière de jeux. Après avoir conçu Zap'T'Balls et Cauldron 3 (testé dans ce numéro), Elmar (c'est son prénom) s'attaque maintenant à Prehistorik 2...



Level 1 : cassez-lui les bras !

Dès le lancement du jeu, on reconnaît la griffe de l'auteur. En effet, il y a le même système qui permet de synchroniser l'écran que dans Super Cauldron. Comme dans ce dernier, le décor est affiché sur un overscan horizontal, et le changement entre les écrans se fait par un scrolling hard, ce qui permet d'avoir un défilement des données ultrafluide. Cette version fonctionne sur tous les CPC munis d'un lecteur disc, et il y a un petit plus, enfin plutôt un énorme plus, pour la version CPC de seconde génération.

C'EST UNE RÉVOLUTION

Oui, c'est une révolution. Souvenez-vous, il y a très longtemps, Ocean avait créé un jeu sur disc qui devait utiliser les possibilités du CPC+, mais le jeu était mal réalisé et l'essai n'avait jamais été renouvelé. Avec Prehistorik 2, ce n'est pas pareil. Les nouvelles possibilités du CPC+ sont exploitées à fond, l'écran est splité, les 4096 couleurs sont bien là et, surtout, les sprites hards sont utilisés d'une manière originale. Ils donnent l'impression, lorsqu'ils sont en mouvement, d'avoir un second plan, et il existe un arrière-plan qui scrolle vers le haut plus lentement que celui où vous jouez. Ainsi le jeu donne une sensation de 3D et un réalisme certain. Dans la version 128 Ko normale, certains conseils sont donnés à l'écran. Ils sont affichés selon un scrolling en raster multicolore avec un tour des lettres sur elles-mêmes.



Une des nombreuses salles secrètes (version 6128).

ET LE SCÉNARIO ?

Le héros, un homme de l'ère préhistorique (on s'en serait douté), doit, au fil des tableaux, se gaver de toutes les denrées qu'il trouvera. Ainsi, il mangera des pommes, des poires ou des oranges, un cornet de frites géant (au look Mc Donald), des hamburgers (quand je vous disais qu'il y avait du fast-food dans l'air) et je ne sais quoi encore, grâce auxquels il attrapera un maximum de bonus, comme des diamants, des vies, des couverts (cuiller, fourchette et couteau), qui transformeront les ennemis en bonus pendant un certain laps de temps, ou des lettres qui serviront à former le mot « BONUS ». Si vous y arrivez, l'ordinateur vous fera crédit de 100 000 points... D'autre part, vous pouvez rebondir sur les ennemis, leur « valeur » sera multipliée par 2, puis par 4, et en rebondissant sur eux en maintenant la direction vers le haut, vous sauterez si haut, que vous pour-

rez attraper des bonus cachés ou des raccourcis. A propos d'ennemis, il en existe des dizaines. Par exemple, dès le premier niveau, vous trouverez des loups-garous, des abeilles, des libellules, des tigres, des tortues géantes, des araignées, des toucans, des dragons, etc. Les boss sont assez gros, et même si cela ralentit l'ensemble, c'est encore jouable (un conseil pour le premier boss, cassez-lui les bras).

ET LA NOTICE ?

Elle est traduite en quatre langues soit l'anglais, le français, l'italien et l'allemand. La personne qui l'a faite n'a pas manqué de glisser quelques petits passages comiques, en voici un extrait :

« N'hésitez pas à frapper un peu partout avec votre arme, si vous voyez une petite fumée se dégager, c'est bon signe. En cherchant bien, vous pourrez peut-être même trouver l'entrée d'un passage secret. Si vous parvenez à faire des signaux de fumée au même rythme que la musique. Vous aurez peut-être la chance de voir apparaître un bison géant qui jouera des claquettes... »

Vous serez récompensé en achetant ce jeu, car lorsque vous aurez bien assimilé son mode d'emploi, vous pourrez afficher dans votre chambre le super poster du jeu que vous trouverez au dos de la notice, cool non ?

SINON, C'EST BEAU ?

Graphiquement parlant, Prehistorik 2 est assez réussi. La version CPC vieille génération a quand même un problème au niveau des couleurs, mais la version Plus, elle, est vraiment très bien faite, la végétation est toujours en mouvement, et les animations ne clignotent que quand les sprites sont très grands (comme les boss de fin de niveau). Bref, je vous encourage vivement à acheter ce soft qui, contrairement à Super Cauldron, est compatible avec les 664. De plus, c'est le premier bon jeu sur CPC+ en disquette à vraiment bien exploiter les possibilités supplémentaires de cette machine (*Nota :* les versions 64 K, 128 K, 128 K pour CPC+ sont



Plus de couleurs sur la version CPC+.



Admirez l'avant-plan sur CPC+.

disponibles sur le même disc). C'est sans conteste l'un des meilleurs jeux existant aujourd'hui sur CPC.

Ludotronic, qui s'étonne toujours des merveilles que peut faire un CPC





Prehistorik 2 is the very last production of Zap'T'Balls and Super Cauldron's creator. Its game-play and its ergonomics make of it one of the best platform games being on our computers. We can only thank Titus for having trusteed Elmar (the programmer) that have converted this hit on CPC.



PREHISTORIK 2 de TITUS

93%

Graphisme :	95%
Son :	85%
Animation :	90%
Richesse :	93%
Scénario :	85%
Ergonomie :	95%
Notice :	95%
Longévité :	95%
Rhaal/Lovely :	98%

CRAZY CARS 3

Fini les circuits de course pour bébés. Moi, je ne prends mon pied qu'en enfreignant la loi du Grand Sam. Qui plus est, il me faut des adversaires à la hauteur...



Dans le magasin.

Tout le monde connaît la légende des crazy cars. Des courses folles sur les autoroutes américaines sans se soucier des grands-pères qui emmènent les petits à la pêche ou le « beauf du dimanche » qui fait tourner le moteur de sa Cadillac, histoire de ne pas la laisser rouiller.

UN MONDE DE FOUS

Ça se passe comme ça au pays des Mac Donald's. A force d'être traités en moutons intelligents (!), les pauvres Ricains se font sauter les plombs. Pour surmonter leurs frustrations quotidiennes, ils doivent s'affronter à travers des jeux qui ne respectent rien. Être, en somme, un de ces légendaires hors-la-loi qui pullulent dans les Westerns au cinéma.

Vous êtes un brave gars à la recherche de sensations fortes. Vous débarquez aux États-Unis et vous placez vos maigres économies dans une Lamborghini Diablo. Vous démarrez dans la vie avec un apport de $6000, et vous rejoignez les quelques fêlés du pays qui participent aux « Saturday Night Races ». Il s'agit de courses de voitures qui ont comme seule règle : être le premier, à tout prix.



Un circuit bien tranquille.

Pour cela, vous devez bien vous équiper et tout faire pour ne pas tomber entre les mains des premiers « cops » venus qui guettent les petits malins comme vous.

Bien entendu, les routes ne seront pas dégagées en votre honneur et grouilleront d'autres voitures qu'il faudra apprendre à éviter.

FLICS, FRIC AND SUN

Je vous le disais plus haut, vous avez au début de votre aventure $6000 en poche. Quelques autoroutes seront à votre disposition. Pour chaque circuit, vous devez payer un engagement qui varie en fonction de la ville où se déroule la course. Celui de Denver, par exemple, est le moins coûteux, mais rapporte, par la même occasion, moins d'argent en cas de victoire. De même, les conditions de courses sont différentes selon les itinéraires. Certaines routes passent par de petits patelins qui, par manque de moyens, ne peuvent se payer une police municipale (ce qui pour tous les concurrents est un avantage non négligeable).

LES PARIS ET LES FRAIS

Au départ de chaque course, parallèlement à votre mise initiale, vous avez la possibilité (en fonction de vos finances) de parier avec les autres concurrents. Il va de soi que ce paquet d'argent rejoindra le portefeuilles du vainqueur de l'étape.

Vous commencez à comprendre que le premier objectif de chaque concur-



La nuit, toutes les Porsche sont grises !

rent, est d'arriver le premier pour empocher le gros lot. Mais à quoi peut bien servir tout cet argent ?
Lors de vos promenades, vous serez amené à érafler la belle carrosserie de votre Diablo. De même, les chocs plus violents pourront endommager votre moteur, voire même rendre votre voiture inutilisable. C'est pour cela, qu'avec de l'argent, vous pouvez passer chez les meilleurs garagistes du coin et, moyennant finances, refaire une santé à votre bolide de rêve.

LES COURSES DE LA COURSE

Les bons pilotes peuvent espérer participer aux plus grands tournois qui ne sont accessibles qu'aux meilleurs. C'est ainsi que votre aventure commencera en quatrième division. Il faudra vous battre pour gravir un à un les différents échelons pour atteindre les troisième, deuxième et, pourquoi pas, première division.
En fonction de votre classement (division) et de votre pécule, vous pourrez acheter (et ce n'est pas donné) les différents équipements qui augmenteront vos chances de réussite. Cela va du simple pneu, boîte de vitesses, détecteur de radar au super turbo-boost ($50.000) en passant par les différents gadgets pour réparer et customiser votre moteur. Il est évident que plus votre voiture sera performante, plus vos chances de réussite seront grandes.
Comme vous le savez, chers lecteurs, *Amstrad Cent Pour Cent* fait tout pour vous donner les petits trucs, cheats, bidouilles ou pokes qui peuvent vous faciliter la vie. J'ai trouvé un petit défaut de programmation dans Crazy Cars 3 qui vous permettra de rouler aux frais de la princesse avec le super turbo-boost de $50.000. Pour cela, allez au magasin et, sans même posséder la somme exigée, achetez le turbo. Bien évidemment, il vous sera refusé mais, lors de la course, appuyez sur la touche Return et regardez votre compteur de vitesse. La classe non ?



Le menu . Vous êtes toujours en 4e division

UNE FOLIE PRENANTE

La réalisation de Crazy Cars 3 est loin de révolutionner le monde de l'animation. Nous avons simplement affaire à un produit correct. Les graphismes sont sympa et riches en couleurs, mais le premier intérêt du jeu réside dans l'ambiance malsaine qui règne chez nos chers Ricains. Vous allez très rapidement comprendre et adopter les règles et style de ce jeu. À ce moment, seules la course et l'envie d'être le premier vous scotcheront à votre fauteuil, mais pour cela, il faudra vous donner la peine d'essayer le produit et ne pas vous fier à vos premières impressions.

Poum



Crazy Cars 3 is certainly the best realization of the trilogy of Titus. Well and good then, he won't oust the masterpiece that is Wec Le-Mans, but it will give a plus to the race and his world of fool that the "Saturday Night Races" make up.







Graphisme :	86%
Son :	70%
Animation :	76%
Richesse :	90%
Scénario :	89%
Ergonomie :	90%
Notice :	80%
Longévité :	86%
Rhaal/Lovely :	92%

SUPER CAULDRON

Super Cauldron... Le rêve de tous les fans de jeux de plates-formes.

Foi de Wolfen, ça faisait un moment que mon CPC n'avait pas eu un bon jeu de plates-formes digne de ce nom à se mettre dans le grille-pain. Il faut dire que ces temps-ci, les amateurs du genre sont à la fête avec la sortie de Super Cauldron et Prehistorik 2 (dont vous n'avez certainement pas raté le test en page 4) commercialisés au même moment. En se penchant sur les deux titres, on remarque d'entrée quelques similitudes comme l'utilisation d'un scrolling hard et d'un overscan horizontal. Et c'est somme toute assez normal, puisque c'est Elmar qui s'est occupé des conversions CPC. Mais voyons de plus près ce qu'il en est de Super Cauldron.

AVERTISSEMENT

Avant de se lancer plus en avant dans la dissection de Super Cauldron, il convient d'apporter une petite rectification quant aux informations de compatibilité de la jaquette. Il s'avère que le logiciel ne tourne pas sur le 664, et c'est vraiment dommage pour les possesseurs de ce CPC (comment ça, je remue le couteau autour de la plaie ?). Enfin, dans un tel cas, vous pourrez toujours vous rabattre sur l'excellent Prehistorik 2 en désespoir de cause.

QUELQUES MOTS SUR L'HISTOIRE

Super Cauldron repose sur un scénario classique, mais d'une richesse peu commune. Le lointain monde baptisé par ses habitants Cauldron est tombé sous la coupe d'un puissant sorcier maléfique. De son château, ce dernier sème la terreur et la désolation, et les effets de sa malédiction se ressentent même sur la flore et la faune qui vous sont devenues hostiles. À l'origine dirigé par de puissants magiciens blancs, le peuple de Cauldron va envoyer une petite sorcière du nom de Zmira afin de tenter de détruire le diabolique sorcier.



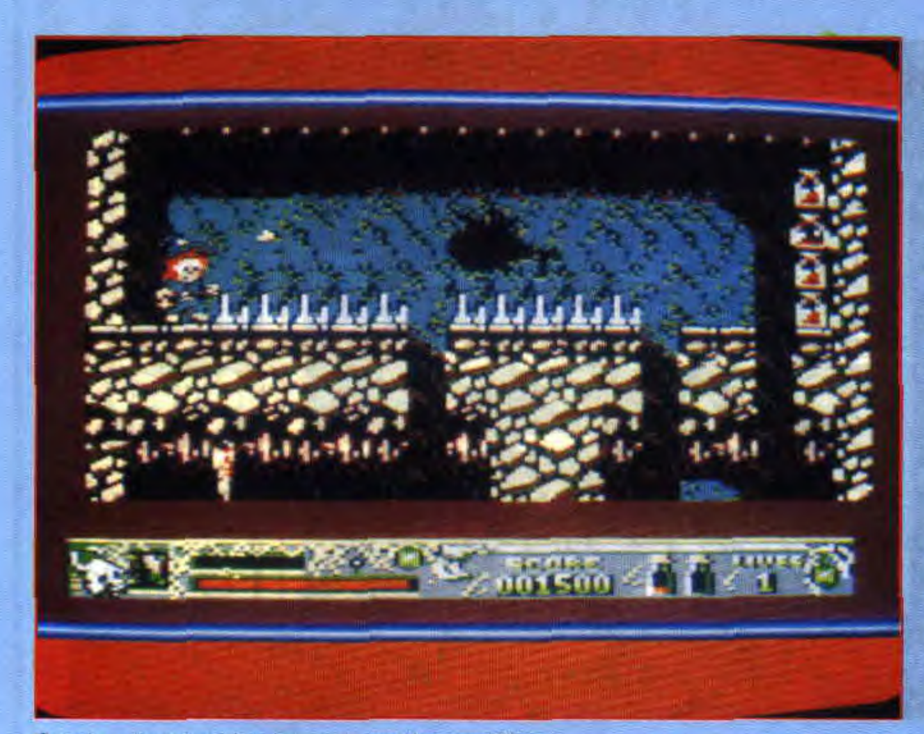
Un scrolling différentiel accompagne les évolutions aériennes de la sorcière.

DÉVELOPPER SES TALENTS DE MAGE

Vous incarnez donc cette petite sorcière, et vous voilà parti pour l'exploration des quatre régions qui constituent le monde de Cauldron. Avant d'affronter le cruel magicien, la petite sorcière va devoir développer ses talents magiques en ajoutant à son grimoire de nouveaux sorts susceptibles de contrer ceux de votre ennemi juré. C'est en tout pas moins d'une douzaine de sorts qui sont répartis un peu partout. Notez que la puissances des différents sorts magiques se dosent en fonction de la durée de pression du bouton de tir. La sélection du sort s'effectue, quant à elle, en consultant votre grimoire. Vous y accédez d'une simple pres-



L'entrée d'une des régions se trouve dans le lit de la rivière.



Quatre potions curatives apparemment inaccessibles.

sion sur la touche Enter En ce qui concerne la progression proprement dite, signalons que chaque monde communique avec des sous-tableaux auxquels on accède par des portes telles ques des souches, des ponts ou d'autres éléments du décor. Dernier point ; dans chaque monde est cachée une clef qui, une fois en votre possession, vous permet d'accéder au niveau suivant. Pour cela, il convient de trouver une salle dans laquelle trône un livre magique qui n'a rien à voir avec les sorts ou votre grimoire. Avant de l'utiliser, il vous faudra venir à bout d'un boss contrôlé par l'esprit de votre maléfique adversaire.

ZOOM SUR LE SCROLLING

Ce test serait incomplet sans consacrer quelques lignes à l'aspect visuel de Super Cauldron. Commençons déjà par souligner le bon choix de la palette de couleurs qui donne un attrait supplémentaire à ce jeu. En plus du scrolling hard qui offre une fluidité peu commune, on dispose également d'un scrolling parallaxe ; ce qui est encore moins courant sur le CPC. C'est indéniable, mais ça saute franchement aux yeux lorsque Zmira collecte les options « balais volants » qui lui permettent, comme à toute bonne sorcière se respectant, de parcourir les cieux.



Ce niveau exige une bonne dextérité.

AUCUN DÉFAUT, OU SI PEU...

Je n'irai pas jusqu'à dire que Super Cauldron n'a aucun défaut mais ils sont tellement rares qu'ils ne méritent pas d'être rapportés. Allez, pour les puristes signalons qu'un petit flash indiquant que l'adversaire est touché par vos projectiles aurait été une bonne chose, mais la gestion des collisions compense largement ce petit manque. Il y a vraiment trop de bonnes raisons pour se procurer dans les plus brefs délais Super Cauldron qui, répétons-le, est de superbe facture. Vous n'avez vraiment pas d'excuses si vous ne l'achetez pas, même s'il est vrai que le choix est dur avec Prehistorik 2. Cela dit, si vous avez les moyens, n'hésitez pas, prenez les deux.

Wolfen



After a difficult time, CPC owners have at last a brilliant platform game to play with. Let's observe that at the same time Super Cauldron is sold, another very good game is coming on the market : Prehistorik 2 ! Based on a scrolling hard use, Super Cauldron offers a flexible animation and the richness of its screenplay is really out of the ordinary. This is game you have to get a matter of urgency.



SUPER CAULDRON de TITUS (6128)

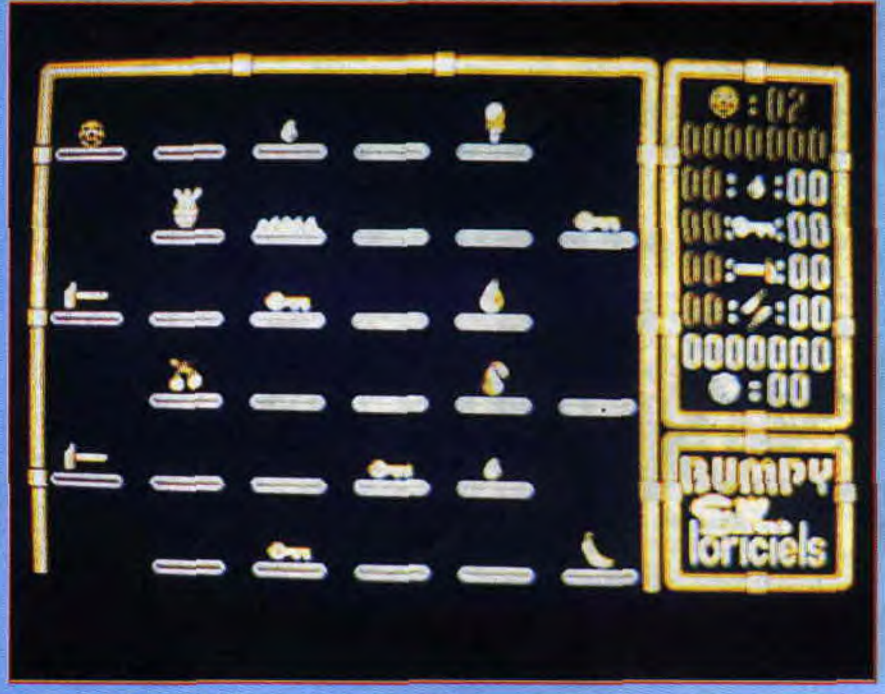


84%

Graphisme :	85%
Son :	80%
Animation :	90%
Richesse :	85%
Scénario :	95%
Ergonomie :	85%
Notice :	75%
Longévité :	80%
Rhaal/Lovely :	80%

MEGA COMPIL

Je ne vanterai pas les mérites pécuniaires et ludiques qu'offrent les compil's, cependant; sachez chers amstradiens que vous ne pouvez pas rater la Mega Compil de Loriciel pour de nombreuses raisons...



Bumpy's, un des chefs-d'œuvre de cette compil'.

À l'heure où nombre d'éditeurs osent abandonner lâchement notre bon vieux CPC, où beaucoup de revendeurs ne s'occupent plus vraiment de notre Amstrad et où beaucoup trop oublient notre belle machine, Loriciel vient réconforter, à la fois, notre logithèque et notre porte-feuilles en sortant avec brio la Mega Compil qui regroupe, sur quatre disquettes, vingt-quatre hits qui ne manqueront pas de vous éblouir...

EH OUI, VINGT-QUATRE D'UN COUP !

C'est en effet la première constatation que l'on peut faire : cette compil' offre sur quatre disquettes, vingt-quatre zolis softs. Je m'en vais (non non, pas chez moi) vous les détailler le plus sommairement possible.
Sur la première disquette, vous trouverez Turbo Cup, Sapiens, Skweek et Maracaibo.
Sur la deuxième disquette, c'est Mobileman, Bumpy, Zox 2099, Cobra, Bactron, L'Aigle d'Or et Kya, que vous retrouverez avec une joie infinie.
Sur la troisième disquette, on change un peu de registre, puisqu'elle contient 3D Fight, Infernal Runner, Charly Diams, Mata Hari, Mission et Fusion 2.
Enfin, sur la quatrième disquette se trouvent Billy 2, Billy-la-Banlieue, Mach 3, Le Cinquième Axe, Foot, Tennis 3D et Rally 2. Avouez tout de même que le programme est alléchant, non ? Oh, ne vous inquiétez pas, nous savons (de Marseille) très bien que votre patience a des limites. Donc, à votre soulagement, il n'est pas question de vous bassiner avec vingt-quatre tests. Pourtant certains softs l'auraient mérité !

UN TOUT PETIT BILAN

Eh oui, un tout petit bilan, parce qu'il me reste encore quelques petites choses à vous dire. Tout d'abord sachez que la Mega Compil n'est au fait qu'un regroupement des compil's Megapack 1 et 2 et La Compil extraordinaire. Ainsi donc, il s'agit d'une compilation de compilations (combien de fois ai-je donc écrit ce mot ?). Enfin de nombreux jeux aurait pu être tout simplement effacés de cette liste, mais, malgré ces quelques petits parasites, de nombreux hits vous attendent. Vous les attendiez ? Alors, que demande le peuple ? Vous ne pouvez plus dire qu'on ne s'occupe pas de vous, hein !

Totov, tout retourné

MEGA COMPIL de LORICIEL : compatible CPC/CPC+



FRANCK ET LES PETITS BATEAUX

Aaaah ! L'été, la plage, les nanas, le soleil, la bronzette, les bikinis tout riquiquis. Quelle chance vous avez, vous qui n'avez pas à vous préoccuper d'une montagne de courrier. Mon château hanté pour un bout de plage.



Comme vous le savez, il en est un, qui répond toujours présent quand le bug s'installe, qui est toujours là pour remettre dans la bonne voie le programmeur égaré, qui ne ménage jamais son double cerveau, au risque de le faire exploser, pour trouver un algorithme vicieux. Cet homme, au sens de l'honneur si développé, ne part jamais en vacances, pour vous permettre d'assouvir votre soif de connaissances.
Cet homme, c'est moi, Francky, Grand Oracle, Maître des Clefs, Protecteur de la Bible du CPC, Souverain du Royaume des Lutins, Grand Gardien de la Vérité cépécienne. Les présentations terminées, on y va.

Quelles sont les instructions les plus utilisées en assembleur et à quoi servent-elles ? Quels sont les noms de tous les registres du Z80 ? et à quoi servent-ils en général ?

Un lecteur curieux

Sans nul doute, les instructions les plus utilisées en LM sont les instructions opérant sur les registres et les transferts mémoire/registre et registre/mémoire, ainsi que les sauts conditionnels ou non, et les instructions arithmétiques sur les entiers. En fait, on ne peut pas citer toutes les instructions les plus utilisées, alors que les instructions les moins utilisées sont facilement listables. DAA fait une conversion de l'octet dans A en nombre BCD compressé dans les haut et bas nibbles de A.
OTDR, OTIR, INDR et INIR sont des instructions d'entrée/sortie qui ne fonctionnent pas sur le CPC, du fait du mauvais câblage du Z80 concernant les entrées/sorties.
Les registres de données sont BC DE HL, il sont 16 bits, mais ils peuvent être scindés en deux registres autonomes de 8 bits : B et C ; D et E ; H et L. Les registres A et F peuvent être associés, mais jamais utilisés ensemble, car F est le registre des flag, c'est-à-dire que chacun de ses bits représente un résultat de test. Ces 8 registres sont doubles, mais on ne peut y accéder en même temps, et, de toute façon, les doublons contiennent des données nécessaires au système lors des connexions des Ram et des Rom. Le registre PC (16 bits) est le Program Counter, il pointe en permanence sur l'instruction qui est exécutée. Le registre SP pointe sur la pile système, quand vous faites un PUSH, les registres sont empilés à l'adresse de SP, et SP est doublement décrémenté. Lors d'un POP, SP est doublement incrémenté et les registres dépilés. Il reste les registres d'indexation IX et IY. Ils sont en général associés à une valeur de décalage (IX + *n*). Cela permet de lire les données dans une table, en ayant les registres d'index mobile (*n*) et le pointeur intra-tableau (IX ou IY) fixe. Voilà, c'est tout pour les registres du Z80.

Salut Francky,
Comme ton domaine de prédilection est l'assembleur, je me permets de te poser ces petites questions. Comment fait-on une boucle en assembleur ? Et comment faire des multiplications avec une constante ?

Charlot

Les boucles, comme les tests, sont les structures les plus importantes dans n'importe quel langage, quel qu'il soit. En langage machine, il y a plusieurs types de boucles : les boucles inconditionnelles, les boucles conditionnelles et les boucles à compteur de boucles.
Bien sûr, on peut combiner ces différentes méthodes pour avoir des types de boucles plus complexes. Il est à signaler qu'une boucle inconditionnelle amène un blocage du programme s'il n'y a pas de saut conditionnel à l'intérieur de la boucle pour en sortir, sinon ça boucle à l'infini.
Les boucles inconditionnelles sont simplement constituées d'un JP ou d'un JR en fin de boucle. Bien sûr, le saut s'effectue vers le début de la boucle. Dans les boucles conditionnelles, le saut ne s'effectue que quand une condition est remplie ou non, l'instruction de saut est conditionnelle, JR NZ,DEBUT ou JP NZ,DEBUT et leurs variantes. Les boucles à compteur commencent par une initialisation d'un registre - qui servira de compteur durant la boucle - avant même le début de la boucle. Si ce registre est le registre B, l'instruction de fin sera DJNZ DEBUT, si c'est un

COURRIER

autre registre, il faudra décrémenter ce registre et faire un saut au début de la boucle si ce registre est différent de zéro, ce qui donne :

```
       LD    a,10       initialisation
Boucle                  début de la boucle
       PUSH  AF         sauvegarde du
                        compteur
       -                instruction a répéter
       -
       POP   AF         rechargement du
                        compteur
       DEC   A          décrémentation
       JR    NZ,boucle  saut si différent de
                        zéro
```

Pour faire des multiplications avec une constante en assembleur, par exemple x 5, il faut faire des multiplications par 2 et des additions de résultats intermédiaires. Pour les multiplications par 2, c'est facile, il suffit de faire un décalage vers la gauche d'un bit, pour une multiplication par 4, il faut faire deux décalages vers la gauche, voilà pour les multiplications avec des puissances de deux. Pour les multiplications avec des nombres impairs, il faut « pusher » des résultats intermédiaires et les réadditionner plus tard. Exemple avec une multiplication de HL par 7 : HL x 7 est égal à HL x 4 + HL x 2 + HL x 1 donc :

```
PUSH   HL      sauvegarde
ADD    HL,HL   multiplication par 2
PUSH   HL      push du facteur 2
ADD    HL,HL   multiplication par 4
POP    DE      récupération du facteur 2
ADD    HL,DE   donne HL x 6
POP    DE      récupération du facteur
               unitaire
ADD    HL,DE   donne HL x 7
```

Ce qu'il faut faire avant tout, c'est voir les puissances de 2 contenues dans le nombre multiplicateur. Le meilleur moyen de le voir, c'est d'afficher le nombre avec la fonction BIN$ du Basic, comme ça vous pourrez préparer le nombre d'additions/décalages et voir quels sont les résultats intermédiaires à sauvegarder et à additionner.

Salut Franck,
Pour commencer, j'adore votre magazine. Cela dit, j'ai un problème, je possède un 6128 moniteur 644, et une GX-4000, mais celle-ci ne se branche que sur des moniteurs CM14 ou MM12. Pourrais-je brancher ma GX-4000 sur mon moniteur de 6128. Si oui, comment ?

Fabrice

Tout d'abord, avant de répondre à ta lettre, je précise à tous que MON adresse c'est :
LE COURRIER DE FRANCKY
MÉDIA SYSTÈME ÉDITION
19, rue Louis-Pasteur
92513 BOULOGNE CEDEX
Eh bien, mon cher Fabrice, il est tout à fait possible de brancher une GX-4000 sur un moniteur CPC ancienne génération, car les signaux sont identiques, mais tu devras fabriquer un petit adaptateur, car les connecteurs sont différents, et il faudra penser à une sortie pour les haut-parleurs externes, car les vieux CPC n'ont pas de haut-parleur intégré au moniteur. De toute façon, ce montage n'a rien de compliqué, mais il vaux mieux qu'il soit fait par un spécialiste, même s'il ne s'agit que de fils à souder ensemble, car, si tu n'as pas les plans, il te sera difficile de le faire tout seul.

Monsieur,
Fidèle lecteur d'Amstrad 100 %, j'ai un problème dans le listing Logon's Demo du n° 46, ligne 160. Il m'affiche Type Mismatch, alors que la ligne ne comporte aucune erreur : 160 READ b$:IF b=VAL("&"+b$) THEN 180. N'auriez-vous pas oublié de tester le listing avant de le publier ?

Nicolas

Mon cher Nicolas, nous n'oublions jamais de tester les listings avant de les publier, c'est une règle que nous ne transgressons jamais. Le Type Mismatch provient du READ B$ qui sert à lire les données stockées dans les lignes de DATA, l'erreur vient justement d'un DATA que le READ n'a pu lire correctement, donc je suggère d'aller voir du côté des DATA où une virgule s'est peut-être transformée en point ou a simplement disparu.

Bonjour Franck,
Un petit mot pour t'encourager, toi et toute l'équipe pour le journal que vous produisez. Dis-moi : peut-on brancher une imprimante HP Thinkjet sur un CPC 6128 ?

Thierry

On peut brancher n'importe quelle imprimante sur un CPC à partir du moment où elle dispose d'un connecteur parallèle Centronic et qu'elle émule le standard Epson. D'après ce que tu me dis, ton imprimante ne dispose que d'un connecteur série (RS 232c), ce qui signifie qu'elle ne pourra même pas être connectée directement au CPC, donc, réserve ton imprimante pour un ordinateur qui pourra la piloter.

Cher Franck,
À quoi servent tous ces DATA dans vos listing ? Pourrais-tu faire un tout petit petit cour sur le POKE ? Merci d'avance et salue tout le monde de ma part.

Heddy

DATA est une pseudo instruction du Basic, pseudo dans le sens où elle n'est pas exécutée. Elle sert à stocker des nombres ou des caractères, voire des phrases ou ce qui y ressemble. Derrière le DATA, il y a des données qui sont séparées par des virgules ou des espaces, si ces données sont des chaînes alphanumériques.
Le POKE sert à mettre une valeur dans une case de la mémoire. La mémoire est comme une feuille de papier quadrillé, où chaque carré contient un nombre appelé octet qui représente une valeur, un chiffre, un caractère, ou une instruction. Chacun de ces carrés a un numéro, l'adresse. En bref, POKE sert à mettre un octet dans une adresse.

Cher Franck,
Moi, Storm-Fire te prie de te pencher sur mes questions restées sans réponses jusqu'à ce jour. J'ai entendu parler d'un certain « CD Games Pack » de Code Masters, il y a bien longtemps. Mais depuis, plus rien. Pourrais-tu m'en donner des nouvelles ? Auparavant, il existait un club à Amstrad Cent Pour Cent, qu'est-il devenu ? Et comment se porte-t-il ? Merci à toi.

Dessert-Storm

Code Master avait créé un CD pour les CPC, mais il semblerait que cette tentative de création d'un nouveau support de jeux pour CPC ait échoué, car tous les cépécéistes ne disposent pas obligatoirement de lecteur de CD. Les données et programmes passant sous forme audio par la connexion cassette, il suffisait d'un simple lecteur CD du commerce pour lire les CD de jeux sur un CPC.
En fait, le club n'a pas disparu, il s'est simplement agrandi à la totalité des lecteurs abonnés, donc le club va bien, merci.

Salut Franck,
Je dois d'abord te féliciter, car ton journal est mon magazine préféré, et ta rubrique est super. C'est pourquoi, je te pose cette question. Il semblerait que l'on puisse programmer en assembleur avec les utilitaires SID, RMAC, et ASM fournis avec le CPM+ (bien que cet assembleur soit différent de DAMS ; par exemple, LD DE,#C000 devient LX D,#C000). Alors peut-on réellement programmer avec eux ou est-ce du domaine de l'utopie ? À propos, ce serait bien de faire une initiation au CPM+.
Sur ce, au revoir et merci.

The Provo-Khator

Bien sûr que l'on peut programmer en LM avec les utilitaires du CPM+, mais le problème, c'est que les programmes sont compatibles avec le microprocesseur 8080 d'Intel qui est le papa de notre cher Z80 fabriqué par Zilog. Le 8080 c'est un Z80 un peu moins évolué, il n'a qu'un seul plan de registres (donc EXX à la poubelle) et n'a pas de registre d'index (donc LD A,[IX+#02] à la poubelle aussi) il lui manque encore d'autres instructions existant sur le Z80. Donc, on peut programmer avec le CPM+, mais ça ne donnerait pas grand-chose.
Une initiation au CPM+... depuis Discology, le CPM+ ne sert vraiment plus à rien, car c'est un système vraiment

trop vieux et obsolète pour penser à en faire une initiation.

Hello Francky
Je t'écris, car je sais que toi seul pourras répondre aux questions d'un pauvre débutant en informatique. Donc voilà. Comment une disquette contient-elle des programmes ? Et puis, à quoi sert la Rom et la Ram ?

Brean Silver

Une disquette, c'est un support magnétique au même titre qu'une cassette audio ou vidéo. Tout support magnétique est capable de mémoriser un champs magnétique créé par une bobine (aimant) et de recréer à la demande ce champ magnétique dans une bobine de lecture. Ce qui crée dans cette bobine un faible - mais suffisant - courant électrique qui peut être géré par le lecteur. Un programme est constitué d'instructions, chacune des instructions est représentée dans la mémoire de l'ordinateur par un code - un nombre -, ces codes sont manipulés sous leur forme binaire, ce qui veut dire que le nombre ne comporte que les chiffres 0 ou 1. Un support magnétique informatique ne peut mémoriser que deux états magnétiques, champs magnétique orienté vers le nord ou vers le sud (vue de l'esprit). L'état orienté vers le nord correspond au 0 et l'état orienté vers le sud au 1. Avant le formatage, une disquette vierge a une magnétisation de sa surface magnétique aléatoire, le formatage permet de créer des sillons qui contiendront les codes des programmes. Ces sillons s'appellent des pistes, et chacune de ces pistes est découpée en secteurs. Sur une disquette, il y a un petit trou dans le disque magnétique qui permet de se positionner exactement sur le premier secteur d'une piste.
La Rom contient des programmes obligatoires pour pouvoir lire une disquette ou une cassette, afficher des trucs à l'écran, traiter des programmes Basic. La Ram est une mémoire à tout faire, elle peut contenir des programmes Basic, la mémoire d'affichage, des données pour les programmes, mais aussi pour le système (contenu en Rom).

Salut Franck
J'ai fait un gros programme en Basic, il pèse dans les 38 Ko, mais quand je le recharge, il me fait Memory Full. Comment ça se fait ?

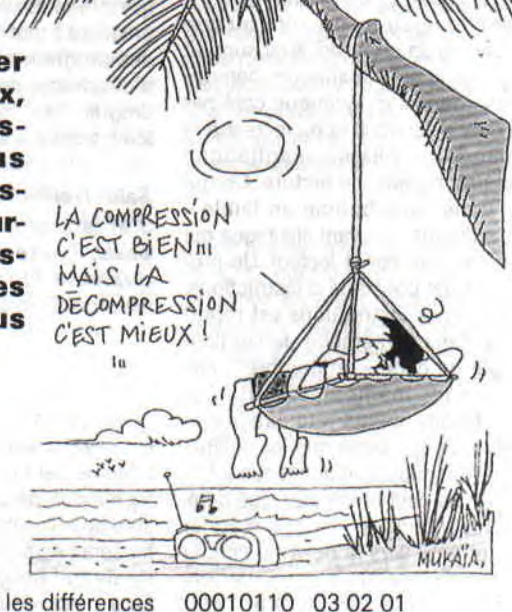
Un lecteur déprimé

C'est en effet un gros problème. Ce qui se passe, c'est que le programme est devenu tellement gros que le système ne peut plus se réserver un buffer tampon de 2 Ko dans la mémoire libre pour la lecture, parce que ce buffer serait écrasé par les deux derniers Ko de ton programme. Ce qu'il faut faire, c'est réduire la taille du fichier dans le header programme sur la disquette avec Disco. Bien sûr, tu vas perdre un bout de programme, mais ça te permettra quand même de le recharger et de le modifier de sorte qu'il soit plus court et rechargeable. Deuxième solution, transfère-le sur cassette avec Disco ou Bsave du CPM 2.2 et modifie-le sur 464. Avec les 464, il y a plus de mémoire disponible grâce à l'absence de certains buffers réservés à la gestion du lecteur de disquette. Bien sûr, il faut que le 464 n'ait pas de lecteur de disquette, sinon les buffers seront présents et ça ne servirait à rien. Le fait de pouvoir sauvegarder un gros programme en Basic ne signifie pas qu'il pourra être rechargé, car, lors de l'écriture, le buffer est plus petit que pendant la lecture, donc à partir de 36 Ko, il vaut mieux s'arrêter, et voir si on ne peut pas optimiser certaines routines pour qu'elles prennent moins de place dans le programme. Par ailleurs, faites attention à vos données. De temps en temps, il vaut mieux les placer dans un fichier plutôt que dans la mémoire, comme ça le programme sera à ses aises et n'en tournera que plus vite si les accès à ces données ne sont pas permanents. Si besoin est, lisez le fichier de données en début de programme pour mettre ces données dans des variables dimensionnées, plutôt que de mettre ces données dans des DATA. Dans tous les cas, n'oubliez jamais que l'espace dont vous disposez en Basic n'est que de 36 Ko utilisables réellement.

Sur ces bonnes paroles, je vous laisse. Bonne bronzette, couvrez-vous bien, et n'oubliez pas votre vieux Franckie...



COMPRESSE DES COMPRESSES



Les images peuvent représenter un suprême plaisir pour les yeux, et un désarroi pour les disquettes. Dix-sept Ko par image nous donnent neuf images par face de disquette, ce qui n'est pas beaucoup. Pourquoi ne pas se faire un petit compresseur-décompresseur spécialisé dans les écrans ? C'est le programme que nous vous proposons dans ces pages.

L'assembleur est le langage le plus rapide que notre micro puisse utiliser. Ça réside simplement dans le fait qu'il n'a aucun travail d'interprétation à réaliser. Il exécute sans se poser de questions. Seulement voilà, la moindre erreur se termine par la même rengaine : le plantage pur et simple. Il faut donc bien réfléchir à l'organisation du programme avant de commencer, histoire de ne pas avoir de surprise. Tracer sur papier évite de se prendre le chou à travailler sur micro. Le cerveau est le meilleur débugger qu'on puisse trouver. Ne l'oubliez jamais.

STRUCTURE DES IMAGES

Si on veut compresser efficacement des images, il faut avant tout regarder de près leur structure. La première chose qui saute aux yeux lors de l'analyse est le fait que d'une ligne à l'autre, il n'existe pas de grandes modifications d'état. Une maison ou un arbre au milieu de l'écran montrent que des données identiques se retrouvent d'une ligne à l'autre. On peut ainsi en déduire la forme de codage suivante par ligne :

- *n* bits représentant les octets à coder ;
- *z* octets ne codant que les valeurs différentes entre les lignes.

Le format d'un fichier compressé d'écran sera donc formé de :

- 80 octets formant la première ligne (la ligne de référence) ;
- 10 octets (10 x 8 = 80 bits) formant le masque de la nouvelle ligne ;
- *z* octets contenant les différences entre les deux lignes.

Ces deux dernières lignes sont codées 199 fois, car 199 lignes sont à reconstruire.

CODAGE DES DONNÉES

Imaginons le codage du sprite de 8 x 4 octets suivant :

```
00 01 02 03 02 01 00 00
00 01 03 02 03 01 00 00
00 01 03 03 03 02 01 00
00 01 03 02 03 02 01 01
```

Voici comment sera préparée la première étape du fichier compressé :

```
Masque     Ligne
           00 01 02 03 02 01 00 00
00111000   00 01 03 02 03 01 00 00
00010110   00 01 03 03 03 02 01 00
00010001   00 01 03 02 03 02 01 01
```

Remarquez que dans le masque (l'octet de gauche donné en binaire) un 1 symbolise une différence entre deux octets d'une même colonne et le zéro signifie que les données sont égales d'une ligne à l'autre, encore dans une même colonne. Lors de l'étape suivante, nous retirons les données redondantes et ne gardons que les octets différents. En voici le résultat.

```
Masque     Ligne
           00 01 02 03 02 01 00 00
00111000   03 02 03
00010110   03 02 01
00010001   02 01
```

De 32 octets, nous passons à 19, ce qui n'est pas négligeable. La décompression se passe sans le moindre problème. Il suffit d'analyser le masque et selon le bit rencontré, s'il vaut 1, on prend un nouvel octet dans la liste et dans le cas contraire, on recopie simplement l'octet correspondant de la ligne précédente. Notez qu'un écran est simplement 10 fois plus large que notre sprite, ce qui ne change rien à l'idée de programmation.

RUSE DE PROGRAMMATION

Dans ces deux routines, nous employons une grosse ruse de programmation consistant à former des instructions de test ou de modification de bits directement dans le code. Ce sont les instructions RES, SET ou BIT qui sont directement codées par la routine, ceci au niveau des labels SUICP des deux programmes. Le reste n'est que broutille. Une routine de création des adresses des lignes écran, une de chargement d'écran, une de traitement et une d'écriture. Rien de sorcier ! Il est possible d'augmenter encore le taux de compression en forçant le CPC à trouver la ligne écran la plus ressemblante à la ligne de référence, mais ceci est une autre histoire...

Sined le Barbare

```
          ORG     #8000
          ENT     $
TABADR    EQU     #A000
TAMPTR    EQU     #5000
ECRAN     EQU     #C000
;
CREADRES  LD      HL,#B800
          LD      DE,#C050
          LD      B,200
          LD      IX,TABADR
CREAD     LD      A,8
          ADD     A,H
          LD      H,A
          JR      NC,RANGAD
          ADD     HL,DE
RANGAD    LD      (IX+0),L
          LD      (IX+1),H
          INC     IX
          INC     IX
          DJNZ    CREAD
;
LOADIMG   LD      HL,NOM
          LD      B,NOM2-NOM1
          LD      DE,ECRAN
          CALL    #BC77
          LD      HL,ECRAN
          CALL    #BC83
          CALL    #BC7D
;
INIT      LD      HL,ECRAN
          LD      DE,TAMPTR
          LD      BC,#50
          LDIR
          PUSH    DE
          POP     IX
          LD      IY,10
          ADD     IY,DE
          LD      C,0
;
GETAD     LD      HL,TABADR
          LD      A,C
          SLA     A
          JR      NC,GETAD1
          INC     H
GETAD1    LD      L,A
          LD      E,(HL)
          INC     HL
          LD      D,(HL)
          INC     HL
          PUSH    DE
          LD      E,(HL)
          INC     HL
          LD      D,(HL)
          POP     HL
          LD      A,#7E
          LD      (SUICP+3),A
          LD      B,80
;
CPLIG     LD      A,(SUICP+3)
          OR      #40
          ADD     A,8
          JR      NC,METOP
          INC     IX
          LD      A,#86
METOP     AND     #BF
          LD      (SUICP+3),A
          LD      A,(DE)
          CP      (HL)
          JR      Z,SUICP
          LD      (IY+0),A
          INC     IY
          LD      A,(SUICP+3)
          OR      #40
          LD      (SUICP+3),A
SUICP     RES     0,(IX+0)
          INC     HL
          INC     DE
          DJNZ    CPLIG
          PUSH    IY
          POP     IX
          LD      DE,10
          ADD     IY,DE
          INC     C
          LD      A,C
          CP      199
          JR      NZ,GETAD
;
SAVING    LD      HL,NOM1
          LD      B,NOM2-NOM1
          LD      DE,#9800
          CALL    #BC8C
          XOR     A
          PUSH    IY
          POP     HL
          LD      DE,TAMPTR
          SBC     HL,DE
          LD      (NOM),HL
          EX      DE,HL
          LD      HL,TAMPTR
          LD      BC,TAMPTR
          CALL    #BC98
          CALL    #BC8F
;
FIN       RET
NOM       DM      DESSIN2A.BIN
NOM1      DM      DES2.BIN
NOM2
;
```

```
          ORG     #8000
          ENT     $
TABADR    EQU     #A000
TAMPTR    EQU     #5000
ECRAN     EQU     #C000
;
CREADRES  LD      HL,#B800
          LD      DE,#C050
          LD      B,200
          LD      IX,TABADR
CREAD     LD      A,8
          ADD     A,H
          LD      H,A
          JR      NC,RANGAD
          ADD     HL,DE
RANGAD    LD      (IX+0),L
          LD      (IX+1),H
          INC     IX
          INC     IX
          DJNZ    CREAD
;
LOADIMG   LD      HL,NOM
LONG      LD      B,NOM1-NOM
          LD      DE,TAMPTR
          CALL    #BC77
          LD      HL,TAMPTR
          CALL    #BC83
          CALL    #BC7D
;
INIT      LD      HL,TAMPTR
          LD      DE,ECRAN
          LD      BC,#50
          LDIR
          LD      IX,TAMPTR+#50
          LD      IY,TAMPTR+#5A
          LD      C,0
;
GETAD     LD      HL,TABADR
          LD      A,C
          SLA     A
          JR      NC,GETAD1
          INC     H
GETAD1    LD      L,A
          LD      E,(HL)
          INC     HL
          LD      D,(HL)
          INC     HL
          PUSH    DE
          LD      E,(HL)
          INC     HL
          LD      D,(HL)
          POP     HL
          LD      A,#3E
          LD      (SUICP+3),A
          LD      B,80
;
CPLIG     LD      A,(SUICP+3)
          ADD     A,8
          CP      #86
          JR      NZ,METOP
          INC     IX
          LD      A,#46
METOP     LD      (SUICP+3),A
SUICP     BIT     0,(IX+0)
          JR      Z,SUIT
          LD      A,(IY+0)
          INC     IY
          JR      META
SUIT      LD      A,(HL)
META      LD      (DE),A
          INC     HL
          INC     DE
          DJNZ    CPLIG
          PUSH    IY
          POP     IX
          LD      DE,10
          ADD     IY,DE
          INC     C
          LD      A,C
          CP      199
          JR      NZ,GETAD
;
FIN       RET
NOM       DM      DES2.BIN
NOM1
;
```

```
10 MEMORY &3FFF:LOAD"COMP.BIN",&8000
20 MODE 2:CAT
30 INPUT "NOM DE L'ECRAN A COMPACTER";N
$:IF N$="" THEN 20
40 FOR I=1 TO LEN(N$):POKE &80C1+I,ASC(
MID$(N$,I,1)):NEXT
50 POKE &8023,LEN(N$):CALL &8000
60 MODE 1:PRINT PEEK(&80C2)+PEEK(&80C3)
*256;" OCTETS DE LONG"
```

```
10 MEMORY &3FFF:LOAD"DECOMP.BIN",&8000
20 MODE 2:CAT
30 INPUT "NOM DE L'ECRAN A DECOMPACTER"
;N$:IF N$="" THEN 20
40 FOR I=1 TO LEN(N$):POKE &8094+I,ASC(
MID$(N$,I,1)):NEXT
50 POKE &8023,LEN(N$):CALL &8000
60 CALL &BB06
```

LE TRI SERA TOP

Ce monstre de la préhistoire informatique va bientôt finir sur vos claviers. En fait, les tris sont certainement les algorithmes les plus utilisés dans les programmes. Qu'il s'agisse de 3D, de gestion de fiches, de scores... dans tous les cas, nous aurons besoin de tris, rudimentaires ou non.

Nous avons déjà parlé de tris. Qu'ils soient à bulles ou par insertion, ils demeurent des tris peu évolués. Les quelques améliorations que ces algorithmes ont vu arriver sur leurs plates-bandes sont le tri « Shell », que nous n'aborderons pas dans ces pages et le tri rapide (dit Quick sort). Si nous ne parlons pas du Shell sort, c'est parce que son mode de fonctionnement est très proche de celui du tri rapide et que ses performances sont, malheureusement pour lui, moins grandes. Nous passerons donc directement au second, sans plus nous poser de questions. C'est M. Hoare qui, en 1962, a mis au point cet algorithme, alors n'oublions pas d'adresser pour lui une petite prière tous les soirs avant de nous endormir.



ERRARE SINEDUM EST

Avant d'entrer dans le vif du sujet, je tiens à vous faire mes plus plates excuses concernant le listing de cette rubrique publié dans le dernier numéro. En effet, s'il n'était pas si laid, il n'avait aucune chance de fonctionner dans l'état dans lequel nous vous l'avions proposé. Notez qu'à cet effet, je me suis fait traiter d'incapable par un lecteur apparemment en mal de pouvoir. Encore pardon ! J'ai présumé de mes connaissances et n'ai pas pris la peine de taper cet exemple. Après trois *ave*, cinq *pater* et 250 pompes en guise de pénitence, je me suis recogné le listing que je me propose de vous rendre dans ce numéro. Merci de votre patience et de votre compréhension.

Message perso : Je ne te salue pas, Jacques du Boulou.

Voici ci-dessous le listing tant convoité.

```
10 DIM TABL$(100)
20 DIM INDEX%(100)
30 FOR I% = 1 TO 100
40 REM Initialisation de l'index
50 INDEX%(I%) = I%
60 NEXT I%
70 REM Remplissage à votre gré de Tabl$()
80 REM En lisant un fichier texte
90 REM Et en le stockant dans Tabl$(), par exemple
100 REM Départ du tri
110 FOR I% = 2 TO 100
120 TEMP% = INDEX%(I%)
130    J% = I% - 1
140    WHILE (J% > 0 AND TABL$(INDEX%(J%)) > TABL$(TEMP%))
150       INDEX%(J% + 1) = INDEX%(J%)
160       J% = J% - 1
170    WEND
180    INDEX%(J% + 1) = TEMP%
190 NEXT
200 REM Affichage du tableau trié
210 FOR I% = 1 TO 100
220    PRINT TABL$(INDEX%(I%))
230 NEXT
```

BASIC ET RÉCURSIVITÉ

Si le Basic Locomotive est un langage sympathique, il n'en est pas moins peu puissant, au regard des langages structurés tels que le C, le Pascal ou les Basic évolués qu'on peut trouver sur des machines telles que le PC. Qu'est-ce que la notion de structure ? C'est simplement le fait de pouvoir disposer de variables locales à des procédures ou à des sous-programmes. Lorsqu'on dispose de tels outils, il est possible d'utiliser la récursivité. Cela consiste en l'appel d'une fonction à partir d'elle-même, et sans que les variables utilisées au sein de celles-ci aient besoin d'être protégées ou sauvegardées. Cela offre des possibilités de programmation qu'on ne peut apprécier qu'en les ayant utilisées. De la même manière, cela entraîne un second problème.

En effet, à chaque appel de la procédure ou de la fonction, un nouveau jeu de variables locales est créé sur la pile, ce qui a tendance à la faire gonfler de manière conséquente, si la fonction se rappelle. Dans ce cas, on dit du sous-programme qu'il est récursif.

CPC PASSÉ

Dans le cadre de l'utilisation de fonctions récursives sur CPC, on se heurte

à un problème simple. Ne disposant pas de récursivité, ce langage ne nous facilitera pas la tâche et il nous faudra trouver un moyen de pallier ce problème. Pour cela, nous devons analyser de près la récursion. Voici comment tout doit se passer. Lorsqu'on entre dans la procédure, il nous faut sauvegarder les paramètres passés dans une zone de travail, ce qui nous permettra de les récupérer ultérieurement. À la sortie de la procédure, on peut détruire les données dorénavant inutiles. Comme une zone de travail doit exister pour chaque instance ou appel de la procédure, il est facile de conclure que plusieurs zones de travail temporaires seront utiles. Grâce à cela, nous déduisons que les zones de stockage intermédiaires devront être dans des tableaux. Avant de se lancer dans des travaux complexes, que diriez-vous de taper un petit programme nous permettant de mettre à jour ce type de problème ?

IDÉOGRAMME D'EXEMPLE

Dans notre exemple programmé, nous appelons simplement une sous-routine destinée à être lancée tant que la variable J%, passée en paramètre, contient une valeur inférieure à 10. Si c'est le cas, le programme est réappelé jusqu'à ce que la condition ne soit plus réalisée. Dans ce cas, il faut nettoyer la pile en exécutant le nombre de RETURN correspondant au nombre de GOSUB effectués. Comme vous pouvez le remarquer, ce travail est automatique. Dans notre cas, il nous faut simplement gérer les variables locales en les rangeant dans le tableau RECURS%. APPEL% est là pour rappeler le nombre d'appels effectués à la sous-routine. RECURS%(APPEL%) est donc la variable locale J% relative à l'appel en cours. Attention à bien faire la différence entre J%, variable globale, et RECURS%(APPEL%), variable locale.

```
10 DIM RECURS%(50)
20 J% = 0
30 APPEL% = 0
40 GOSUB 60
50 END
60 APPEL% = APPEL% + 1
70 J% = J% + 1
80 RECURS%(APPEL%) = J%
90 PRINT "Appel n°"; APPEL%;
100 PRINT " * Variable locale ="; J%
110 IF J% < 10 THEN GOSUB 60
120 PRINT "Retour n°"; APPEL%;
130 PRINT " * Variable locale ="; RECUR
S%(APPEL%)
140 APPEL% = APPEL% - 1
150 RETURN
```

Voici le petit programme :

LES TRIS TUERAIENT

Maintenant que ces notions sont bien en place, nous pouvons aborder sans crainte le tri rapide. En voici l'algorithme :

- couper arbitrairement la zone à trier en deux parties ;
- sélectionner le pivot en prenant l'élément de coupure ;
- établir deux pointeurs (un montant à partir du début et l'autre descendant en partant de la fin) ;
- tant que les pointeurs ne se sont pas croisés (point Start) ;
- tant que les éléments du début sont inférieurs au pivot, avancer le premier pointeur ;
- tant que les éléments de la fin sont supérieurs au pivot, reculer le second pointeur ;
- échanger les deux valeurs visées par les pointeurs ;
- retourner au point Start ;
- lancer le tri rapide sur la première demi-zone ;
- lancer le tri sur le reste.

La récursivité nous permet de garder les paramètres de début et de fin de zone, et de les récupérer après que les premiers appels récursifs eurent été réalisés.

EXPLICATIONS PROGRAMMÉES

Dans le cadre de notre programme, garanti 100% sans bug, nous utilisons le tableau a%(element%) pour stocker des valeurs entières. Il est possible de ranger ici toute sorte de données. N'oubliez simplement pas d'utiliser un tableau d'index si vous utilisez des chaînes de caractères. Element% est le nombre d'éléments utilisables dans le tableau. Recs%() et Recf%() sont les tableaux qui stockent respectivement deb% et fin% soit les valeurs intermédiaires des paramètres à passer à la routine de tri rapide. Ces tableaux sont attachés à l'évolution de la variable Appel% qui stocke le nombre d'appels récursifs à la routine. On peut calculer une valeurs approximative du nombre d'appels à l'aide de la formule :

Taille de Recs%() et Recf%() = ln(element%) + 10

Ici la valeur 10 est simplement une marge de sécurité, dans le cas ou les valeurs de pivot ne seraient pas vraiment centrées. Ln est bien entendu la fonction de calcul d'un logarithme népérien. Nous avons forcé ces tailles à 50 de manière à être sûr qu'un débordement de tableau n'altérera pas le déroulement du programme.

TECHNIQUE-MENT VÔTRE

Le Basic Locomotive du CPC est capable d'imbriquer 83 Gosub sans problème. Si on se réfère au calcul du nombre d'appel, cela nous donne quelque chose comme 2^{83} soit 9×10^{24}. Dans le cadre de l'installation de cette routine dans un programme, il faudra bien entendu prendre l'environnement en compte. La pile est bien fragile, surtout lorsqu'on la manipule avec récursivité. Avec tout cela, si vous n'arrivez pas à ranger des données, revendez votre CPC. Notez que si nous avons

```
10 '
20 ELEMENT% = 100
30 DIM A%(ELEMENT%)
40 DIM RECS%(100)
50 DIM RECF%(100)
60 '
70 FOR I% = 1 TO ELEMENT%
80 A%(I%) = RND(1) * 2000
90 NEXT
100 '
110 DEB% = 1
120 FIN% = ELEMENT%
130 GOSUB 210
140 '
150 FOR I% = 1 TO ELEMENT%
160 PRINT A%(I%),
170 NEXT
180 '
190 PRINT : END
200 '
210 ' Tri rapide (deb%, fin%)
220 IF FIN% - DEB% < 2 THEN GOTO 500
230 PIVOT% = A%((DEB% + FIN%) / 2)
240 S% = DEB%
250 E% = FIN%
260 WHILE (S% < E%)
270 WHILE A%(S%) < PIVOT%
280 S% = S% + 1
290 WEND
300 WHILE A%(E%) > PIVOT%
310 E% = E% - 1
320 WEND
330 IF S% > E% THEN 390
340 Z% = A%(E%)
350 A%(E%) = A%(S%)
360 A%(S%) = Z%
370 S% = S% + 1
380 E% = E% - 1
390 WEND
400 APPEL% = APPEL% + 1
410 RECS%(APPEL%) = S%
420 RECF%(APPEL%) = FIN%
430 FIN% = E%
440 GOSUB 210
450 DEB% = RECS%(APPEL%)
460 FIN% = RECF%(APPEL%)
470 APPEL% = APPEL% - 1
480 GOSUB 210
490 RETURN
500 IF DEB% = FIN% THEN RETURN
510 IF DEB% = 0 THEN RETURN
520 IF A%(DEB%) < A%(FIN%) THEN RETURN
530 Z% = A%(DEB%)
540 A%(DEB%) = A%(FIN%)
550 A%(FIN%) = Z%
560 RETURN
```

quelque chose comme quatre Return pour quelques Gosub, c'est dans un souci de rapidité. Repasser par un Goto n'est pas ce qu'il y a de plus optimisé.

C'est ici que finit cet article. Je vous souhaite donc bon tri...

Sined le Barbare

VECTORISER SANS VEXER

Enfin, nous allons toucher au but. C'est dans ces pages que la dernière ligne droite d'énumération de vecteurs va prendre fin. En tête, nous trouvons actuellement « interfaçage avec le matériel » qui mène d'une longueur d'avance sur « gestionnaire sonore ». Lequel des deux va l'emporter ?

Ah, vous vouliez programmer système ! Ah, il vous fallait des informations sur le contenu de la mémoire morte de notre bon vieux CPC ! Eh bien voilà, nous en avons presque fait le tour. J'attends maintenant de vous quelques programmes, utilitaires ou non, qui nous permettront de voir que ces pages n'ont pas été vaines. Le meilleur de vos produits sera publié au sein de ce magazine, alors ne vous faites pas prier. Attaquez extemporanément.

Dans le dernier numéro, nous avions laissé les vecteurs à l'étape du Kernel. Ayant débroussaillé ce noyau, nous voici arrivés à l'interfaçage avec le matériel qui est tout de même assez important. Nous pourrons ainsi entrevoir les capacités de la machine, qu'elles concernent le port imprimante, les interruptions ou encore le contrôleur vidéo. Comme dans les derniers épisodes, CA signifie « conditions d'appel », CF « conditions de fin ». Avec tout ça, vous aurez de quoi faire. Et c'est parti !



INTERFACE MATÉRIELLE

Les programmeurs de la Rom ont classé ainsi tous les gestionnaires de mémoire morte, les compteurs, les points d'entrée d'interruptions... Voici la liste de ces vecteurs :

• BD10 : positionne le compteur interne à une valeur précise

BD10 est utilisé à l'allumage de la machine pour mettre l'horloge interne à zéro.

CA : DE contient le poids fort de la valeur ; HL contient le poids faible.

CF : AF est modifié.

• BD13 : charge un programme en Ram et le lance

Quoi de plus normal qu'un système se charge de cette tâche.

CA : HL contient l'adresse de la routine à appeler pour charger le programme (le vecteur cassette ou disquette accompagné des paramètres adéquats).

CF : les registres modifiés par le programme appelé.

• BD16 : lance un programme d'une Rom de second plan

Il est agréable de disposer de vecteurs indirects permettant d'appeler des routines stockées en Rom.

CA : HL contient l'adresse de la routine ; C représente le numéro de la Rom.

CF : Dis monsieur le programme en Rom, quels registres modifies-tu ?

• BD19 : synchronisation avec le CRTC

Appeler BD19 revient à demander au CPC d'attendre que le balai qui génère l'image écran soit arrivé derrière le moniteur.

CA : aucune.

CF : pas plus, pas moins.

• BD1C : positionnement du mode écran

Vous connaissez l'instruction Basic Mode, alors vous connaissez ce vecteur.

CA : A contient le numéro du mode (0, 1 ou 2).

CF : AF est modifié.

• BD1F : positionnement de l'offset de la mémoire vidéo

On peut se servir de ce vecteur pour opérer des scrollings hardwares à l'écran.

CA : A contient la base de l'adresse de la mémoire vidéo (&00, &40, &80 ou &C0 selon le segment) ; HL contient l'offset (notez que ce dernier ne progresse que par pas de deux).

CF : AF est modifié.

• BD22 : standardisation de la palette

Toutes les encres sont mises dans la couleur précisée (simulation d'effacement d'écran).

CA : DE contient l'adresse sur un vecteur encre (deux couleurs).

CF : AF est modifié.

• BD25 : la palette modifiée *(bis)*

Comme pour le vecteur BD22 mais agit aussi sur le border.

CA : DE contient l'adresse sur un vecteur encre.

CF : AF est modifié.

• BD28 : initialisation du détournement vers l'imprimante
Il y a du PRINT #8 dans l'air.
CA : pas de paramètre.
CF : AF, BC, DE et HL sont modifiés.

• BD2B : envoi d'un caractère à l'imprimante avec retour
Si l'imprimante est occupée, le système fera en sorte d'attendre un peu et rendra la main avec un statut d'erreur si besoin est. Attention ! la donnée envoyée est perdue quoi qu'il arrive.
CA : A contient le code Ascii du caractère à envoyer.
CF : AF est modifié. Si la retenue (carry) est vraie, tout s'est bien passé. Si elle est fausse, l'imprimante est restée occupée trop longtemps.

• BD2E : test de l'imprimante
Le signal testé se nomme Busy.
CA : rien à faire.
CF : si la retenue est vraie, l'imprimante est occupée.

• BD31 : envoi d'un caractère à l'imprimante sans retour
Pour l'appel de cette fonction, l'imprimante ne doit pas être occupée.
CA : A contient le caractère à envoyer.
CF : la retenue est toujours vraie ; AF toujours modifié.

• BD34 : envoi d'une donnée dans le PSG AY3 W8912
Quel nom barbare pour parler du générateur sonore. Si vous voulez en savoir plus, consultez les anciennes rubriques Bidouilles qui se sont étalées sur ce sujet.
CA : A contient le numéro du registre ; C contient la donnée.
CF : AF et BC sont modifiés.

LES BLOCS DE SAUT

Tous les vecteurs sont rangés en mémoire selon la grande liste que nous avons parcourue ensemble. Si certains ont étés détournés à des fins plus ou moins douteuses, il est possible de les réinitialiser grâce au vecteur suivant.

• BD37 : réinitialise les blocs de saut standard
C'est pas un CALL 0 mais presque !
CA : rien à faire, cool !
CF : AF, BC, DE et HL modifiés.

EN AVANT LA MUSIQUE

N'oublions pas que le CPC ne serait pas ce qu'il est s'il ne disposait pas de son fabuleux générateur sonore. Si celui-ci est très pratique pour les bruitages, car autonome une fois programmé, il n'en va pas de même pour la musique. Les générateurs d'enveloppes qu'il utilise sont en effet trop peu malléables pour être utilisés dans de bonnes conditions. C'est pour cette raison que la Rom du CPC contient des primitives de gestion de files d'attente de notes, sons et enveloppes. Tout ceci est piloté à partir des interruptions et permet de pratiquer la musique sans avoir à s'arracher les cheveux. Inutile donc d'être un grand programmeur en assembleur pour entendre ses mélodies sous leur plus belle forme. Bien entendu, l'utilisation d'un tel système met en œuvre de nombreux vecteurs que nous allons passer en revue dans ces pages. Les structures de données utiles seront vues au fil des besoins.

• BCA7 : initialisation du gestionnaire sonore
À appeler pour que cela fonctionne.
CA : rien à préciser.
CF : AF, BC, DE et HL modifiés.

• BCAA : ajoute un son à la queue sonore
Les notes sont disposées dans une liste à la queue leu leu, et il faut bien disposer d'un gestionnaire permettant de les y introduire. Voici la structure des données à envoyer : canal (1 octet), enveloppe de volume (1 octet), enveloppe de ton (1 octet), fréquence initiale (2 octets), bruit initial (1 octet), volume initial (1 octet), durée (2 octets).
CA : HL contient l'adresse du son (il doit se trouver dans les 32 Ko de mémoire centrale).
CF : si la retenue est vrai, le son a pu être ajouté. Si elle est fausse, les queues étaient pleines. Dans tous les cas, HL, BC, DE et IX sont modifiés.

• BCAD : teste si une queue sonore est pleine
Pratique pour savoir si on peut insérer des sons.
CA : A contient le numéro du canal à tester (1 pour A, 2 pour B et 4 pour C).
CF : A contient l'état du canal testé (4 si le canal est vide, 132 s'il joue). AF, BC, DE et HL sont modifiés.

• BCB0 : prépare l'exécution d'une interruption lorsqu'une queue sonore est vide
Il est bon de pouvoir faire un petit truc lorsqu'une musique est finie.
CA : A contient le numéro du canal à tester ; HL contient l'adresse du programme à lancer.
CF : AF, BC, DE et HL modifiés.

• BCB3 : remet les sons en route sur un canal
Il est possible de stopper l'émission de sons à l'aide du vecteur suivant.
CA : A contient le numéro de canal à remettre en fonctionnement (1 pour A, 2 pour B et 4 pour C).
CF : AF, BC, DE et HL modifiés.

• BCB6 : arrêt de tous les sons
Chut !
CA : si un son était émis, la retenue est vraie.
CF : AF, BC, DE et HL modifiés.

• BCB9 : remet tous les canaux en marche
Inverse l'action du vecteur précédent.
CA : rien à faire.
CF : AF, BC, DE et IX modifiés.

• BCBC : initialisation d'une des 15 enveloppes d'amplitude
Comme sous Basic, avec les mêmes paramètres.
CA : A contient le numéro de l'enveloppe ; HL pointe sur les données définissant les amplitudes ; BC contient la longueur des données à prendre en compte.
CF : si la retenue est vraie, l'enveloppe a été initialisée et Hl incrémenté 16 fois. Dans tous les cas, HL, BC de et HL modifiés.

• BCBF : initialisation d'une des 15 enveloppes de fréquence
Remarques inutiles.
CA : A contient le numéro d'enveloppe ; HL pointe sur les données définissant les fréquences ; BC contient la longueur des données à prendre en compte.
CF : mêmes remarques que pour le vecteur précédent.

• BCC2 : fournit l'adresse d'une enveloppe d'amplitude
Il est pratique de modifier les enveloppes dynamiquement en mémoire. Pour cela, rien ne vaut l'adresse des données convoitées.
CA : A contient le numéro d'enveloppe.
CF : si l'enveloppe a été trouvée, la retenue est vraie. Dans ce cas, HL contient l'adresse de l'enveloppe et BC sa longueur. Sinon, HL est modifié et BC préservé. Dans tous les cas, AF est modifié.

• BCC5 : fournit l'adresse d'une enveloppe de fréquence
Pas de commentaire puisqu'on en parle déjà au-dessus.
CA : A contient le numéro d'enveloppe.
CF : si l'enveloppe a été trouvée, la retenue est vraie. Dans ce cas, HL contient l'adresse de l'enveloppe et BC sa longueur. Sinon, HL est modifié et BC préservé. Dans tous les cas, AF est modifié.

LA FIN DES FINS

Vous ne pourrez pas dire qu'on n'aura pas tout fait pour vous donner des informations concernant les vecteurs. Depuis un paquet de numéros, on vous flatte et vous offre le nécessaire à une programmation en assembleur facilitée. Je relance le mini concours. Envoyez vos produits et en échange, ils passeront dans les pages de votre magazine préféré. C'était le dernier épisode de la saga des vecteurs du CPC. Que leur utilisation vous soit propice. En attendant, bonne programmation !

Sined le Barbare

LA SCAROLE VERTE Y CALE...

C'est sous ce jeu de mots douteux que je vais vous parler des scrolls hards verticaux, encore appelés Scrolling-Registre 5 *en raison de l'aide précieuse que fournit ce* registre du CRTC pour décaler vers le haut (ou vers le bas), et avec fluidité, tout un écran.

Cette technique que nous allons vous décrire est employée dans le jeu Mission Genocide (un budget de Firebird que je ne saurais trop vous conseiller) ou dans quelques démos (comme le menu et la Nega-part de The Demo ou la BSC Megademo par exemple).

La méthode est relativement simple (encore fallait-il la trouver), et je m'en va vous l'expliquer de ce pas. Il suffit en fait de séparer l'écran en deux parties au moins, comme pour une rupture toute simple, ainsi que papy Longshot vous l'a expliqué il y a quelque temps dans un précédent numéro. Ce n'est qu'ensuite que le registre 5 intervient : grâce à lui, on décale sans se fatiguer d'une ligne ou plus un des écrans de la rupture, vers le haut ou vers le bas. Comme le registre 5 ne peut varier qu'entre 0 et 31, on ne se sert, en réalité, que des valeurs 0 à 7, et on décale alors l'écran également à l'aide du pointeur Video, autrement dit, les registres 12 et 13 du CRTC. Il faut juste faire attention à une chose, ne pas oublier de changer également la valeur du registre 5 des autres écrans de la rupture. Celle-ci doit être égale au complément à 7 de la valeur du registre 5 de l'écran qui scrolle. Cela doit être *fait pour stabiliser les écrans et éviter* qu'ils ne sautillent.

Voilà pour la partie hard, mais il reste encore la partie soft, autrement dit l'affichage des motifs du scroll. Eh bien, il suffit de descendre, à chaque balayage d'une ligne sur l'écran, si le scrolling est ascendant et réciproquement, afin de compenser le décalage *généré par le scrolling hard.* Sous des aspects compliqués, ce système est en réalité très simple, et il est beaucoup plus rapide qu'un scrolling vertical réalisé en soft. Son seul inconvénient est la nécessité d'une rupture, mais ce problème disparaît avec un usage avisé des interruptions, ce qui est en quelque sorte réalisé dans le programme ci-contre puisqu'il se synchronise sur des HALTs.

Voilà justement une parfaite transition pour vous parler du programme de cette fois-ci, mais je trouve qu'il est suffisamment commenté pour ne pas avoir à l'expliquer ; disons simplement qu'il fait défiler un message à caractères géants grâce à un scrolling hard vertical. Par contre, je vais vous parler de la façon dont est générée la fonte géante du scrolling.

Étant donné que chaque caractère ne prend que 64 octets en mémoire et fait 256 lignes de haut sur 64 octets (256 pixels en mode 1) sur l'écran, soit 16 Ko, je crois que le mot « géant » n'est pas de trop. Nous avons déjà *vaguement parlé de cette technique de* compression lors du listing d'une démo. Chaque octet des caractères en mémoire correspond, en fait, à un morceau du caractère géant, un « bloc » utilisé par plusieurs caractères. Dans le programme, ces blocs sont définis grâce à des caractères Ascii assemblés pour ne pas prendre trop de place. Pour la même raison, il n'y a dans le programme que les matrices des caractères qui composent le mot « 100% ». Toutefois, il y a bien une gestion de texte, c'est-à-dire que vous pouvez très bien vous amuser à définir les matrices des autres caractères, avec une restriction cependant : ces caractères doivent être dans l'ordre Ascii. Voilà pourquoi, dans le programme, le « % » de « 100% » est *remplacé par un « / » (slash), code qui* est juste avant le « 0 » et le « 1 » dans l'ordre Ascii. Je rajouterai enfin que ce type de compression par bloc est utilisé, non seulement, dans les démos pour faire défiler des caractères, mais également dans les jeux pour les scrollings de décor.

Donc, le programme fait défiler l'écran du haut vers le bas à raison d'une ligne par balayage. Je vous laisse le soin de voir comment changer le sens ou la vitesse, ce n'est pas compliqué, mais cela demande rigueur et attention. De toute façon, on ne possède que ce dont on a l'expérience...

Sur ces paroles hautement philosophiques (si ! si !), je vous laisse méditer. A bientôt !

Pict/Logon System

```
;
;         SCROLLING HARD VERTICAL
;
;         (c)Pict/Logon System Mai 1993
;
;         Assemble avec DAMS
;
;         charge en banque #c4 (#4000)
;
          ORG  #3000
          ENT  $
;
;         largeur en mots de l'ecran:
;
r1        EQU  32
;
;         Initialisation mode et texte
;
          LD   a,1
          CALL #bc0e
          CALL #bb4e
;
;         Dessin de la matrice
;         des blocs
;         (6 blocs de 4 caracteres
;         de hauteur)
;
          LD   b,6*4
          LD   hl,tabchr
lpaff1    PUSH bc
          LD   b,4
lpaff2
;
;         on affiche une ligne
;         de chaque bloc
;
          LD   a,(hl)
          INC  hl
          CALL #bb5a
          DJNZ lpaff2
          POP  bc
;
;         on passe au debut de
;         la ligne suivante grace
;         aux codes de controle
;
          LD   a,10
          CALL #bb5a
          LD   a,13
          CALL #bb5a
          DJNZ lpaff1
;
;         On sauve les blocs
;
          LD   b,24*8
          LD   hl,#c000
          LD   de,matrix
lcopy     PUSH bc
          LD   bc,8
          LDIR
          LD   bc,#800-8
          ADD  hl,bc
```

LOGON

```
        JP   nc,nocarry
        LD   bc,#c050
        ADD  hl,bc
nocarry POP  bc
        DJNZ lcopy
;
;       Affiche le message
;       en bas de l'ecran
;
        XOR  a
        CALL #bc0e
        LD   h,15
        LD   l,2
        CALL #bb75
        LD   hl,mess
        LD   e,2
affmess
        LD   a,e
        PUSH hl
        CALL #bb90
        POP  hl
        INC  e
        LD   a,(hl)
        OR   a
        JP   z,messaff
        INC  hl
        CALL #bb5a
        JP   affmess
messaff
;
;       Configure le format
;       de l'ecran
;
        LD   bc,#bc01
        OUT  (c),c
        LD   BC,#BD00+r1
        OUT  (c),c
        LD   bc,#bc02
        OUT  (c),c
        LD   bc,#bd00+43
        OUT  (c),c
;
;       On interrompt le
;       firmware
;
        DI
        LD   hl,(#38)
        LD   (syst+1),hl
        LD   hl,#c9fb
        LD   (#38),hl
;
;       Passe en Banque #c0
;       qui est en fait une
;       partie de l'ecran.
;
        LD   bc,#7fc0
        OUT  (c),c
;
;       Efface l'ecran en #4000
;
        LD   hl,#4000
        LD   (hl),0
        PUSH hl
        POP  de
        PUSH de
        POP  bc
        INC  de
        DEC  bc
        LDIR
;
;       Change les couleurs
;       des encres
;
        LD   hl,mess
        LD   bc,#7f00+13
colorlp OUT  (c),c
        DEC  hl
        LD   a,(hl)
        OUT  (c),a
        DEC  c
        JP   nz,colorlp
;
;       Boucle principale
;
BCLP    DI
;
;       on attend la synchro Verticale
;
        LD   B,#F5
vs      IN   A,(C)
        RRA
        JP   NC,vs
;
;       Aussitot apres,on configure
;       les registres pour la rupture
;
;       On affiche 256 lignes
;
        LD   BC,#BC06
        OUT  (C),C
        LD   BC,#BD20
        OUT  (C),C
        LD   BC,#BC07
        OUT  (C),C
        LD   BC,#BD7F
        OUT  (C),C
;
;       On passe en mode 1
;       et on initialise le
;       diviseur d'interruption
```

```
;
        LD   BC,#7F9d
        EI
;
;       On met en noir l'encre 1
;       afin de cacher l'effet de
;       saccade creer par le decalage
;       de l'ecran
;
        LD   de,#0154
        OUT  (c),c
        OUT  (c),d
        OUT  (c),e
;
;       Gestion Scroll Vertical
;
;       on envoie l'adresse Hard
;       de l'ecran aux registres
;       du CRTC
;
OFFSET  LD   HL,#1000
        LD   BC,#BC0C
        OUT  (C),C
        INC  B
        OUT  (C),H
        DEC  b
        INC  c
        OUT  (C),C
        INC  B
        OUT  (C),L
;
;       On fait monter l'ecran
;       d'une ligne grace au
;       registre 5,et toutes
;       les 8 lignes,on change
;       l'adresse Hard de l'ecran
;
charup  LD   de,r1
VERTY   LD   A,8
speed   SUB  1
        AND  7
        LD   (VERTY+1),A
        JP   nz,ver
        ADD  HL,de
ver
;
;       On s'assure que l'adresse
;       hard de l'ecran scrollant
;       ne deborde pas
;
        LD   c,a
        LD   a,h
        AND  %11
        OR   #10
        LD   h,a
        LD   (OFFSET+1),HL
;
;       On envoie la valeur de
;       decalage au registre 5
;       du crtc
;
        LD   a,c
xor1    XOR  0
        LD   BC,#BC05
        OUT  (C),C
        INC  B
        OUT  (C),A
;
;       on attend quelques lignes
;
        LD   bc,260
tempo
        DEC  bc
        LD   a,b
        OR   c
        JP   nz,tempo
;
;       puis on initialise
;       le reg 4 du CRTC
;
        LD   BC,#BC04
        OUT  (C),C
        LD   BC,#BD1B
        OUT  (C),C
;
;       ainsi que l'adresse
;       du second ecran
;
OFFSET2 LD   HL,#3000
        LD   BC,#BC0C
        OUT  (C),C
        INC  B
        OUT  (C),H
        DEC  b
        INC  c
        OUT  (C),C
        INC  B
        OUT  (C),L
;
;       Il faut alors
;       complementer le
;       registre 5
;
VERTICA1 LD  A,(VERTY+1)
        LD   BC,#BC05
        OUT  (C),C
        INC  B
xor2    XOR  7
        OUT  (C),A
;
```

```
;           puis on affiche
;           le 1er raster
;
            LD    bc,3
temp1       DEC   bc
            LD    a,b
            OR    c
            JP    nz,temp1
            LD    hl,raster1
            LD    a,10
            LD    bc,#7f01
            OUT   (c),c
rastlp1     LD    c,(hl)
            OUT   (c),c
            INC   hl
            LD    e,12
tempr1
            DEC   e
            JP    nz,tempr1
            NOP
            NOP
            DEC   a
            JP    nz,rastlp1
;
;           On a plus qu'a attendre
;
N1          HALT
N2          HALT
N3          HALT
N4          HALT
;
;           ...attendre encore...
;
            LD    bc,324
temp2       DEC   bc
            LD    a,b
            OR    c
            JP    nz,temp2
;
;           et afficher le 2nd raster
;
            LD    hl,raster2
            LD    a,10
            LD    bc,#7f01
            OUT   (c),c
rastlp2     LD    c,(hl)
            OUT   (c),c
            INC   hl
            LD    e,12
tempr2
            DEC   e

            JP    nz,tempr2
            NOP
            NOP
            DEC   a
            JP    nz,rastlp2
N5          HALT
;
;           Passe en mode 0
;
            LD    bc,#7f8c
            OUT   (c),c
;
;           Reconfigure les
;           registres du CRTC
;           pour le 2eme ecran
;
            LD    BC,#BC04
            OUT   (C),C
            LD    BC,#BD07
            OUT   (C),C
            LD    BC,#BC07
            OUT   (C),C
            LD    BC,#BD05
            OUT   (C),C
            LD    BC,#BC06
            OUT   (C),C
            LD    BC,#BD05
            OUT   (C),C
;
;           Gestion de l'affichage
;
;
;           les blocs font 32 lignes
;           de hauteur
;
ctmat       LD    a,#1f
            INC   a
            AND   #1f
            LD    (ctmat+1),a
            JP    nz,cnty
;
;           les caracteres font 8 blocs de hauteur
;
ctchr       LD    a,7
            INC   a
            AND   7
            LD    (ctchr+1),a
            JP    nz,chline
;
;           gestion du texte
;
ctxt        LD    hl,text
            LD    a,(hl)
            INC   hl
            OR    a
            JP    nz,charok
            LD    hl,text
            LD    a,(hl)
            INC   hl
charok
            LD    (ctxt+1),hl
            CP    32
            JP    nz,nospace
;
;           si c'est un caractere
;           d'espacement,on affiche
;           des octets nuls (qui sont
;           en fait dans le 1er bloc)
;
            LD    hl,matrix
            LD    (chline+1),hl
;
;           et on s'arrange pour qu'il
;           fasse 4 blocs de haut
;
            LD    a,4
            LD    (ctchr+1),a
            JP    chline
;
;           sinon gn trouve quelle
;           matrice lui correspond
;           (on multiplie par 64,
;           ce qui est la taille
;           de la matrice d'un
;           caractere)
;
nospace
            SUB   47
            LD    h,0
            LD    l,a
            ADD   hl,hl
            ADD   hl,hl
            ADD   hl,hl
            ADD   hl,hl
            ADD   hl,hl
            ADD   hl,hl
            LD    de,tabchar
            ADD   hl,de
            LD    (chline+1),hl
;
;           on copie l'adresse de la
;           des blocs dans un buffer
;
chline      LD    hl,tabchar
            LD    bc,matrix
            PUSH  iy
            LD    iy,buffer
            LD    a,8
nline
            LD    d,(hl)

            LD    e,0
            EX    de,hl
            ADD   hl,bc
            EX    de,hl
            INC   hl
            LD    (iy+1),d
            LD    (iy+0),e
            INC   iy
            INC   iy
            DEC   a
            JP    nz,nline
            POP   iy
            LD    (chline+1),hl
;
;           On descend sur l'ecran
;           d'une ligne pour compenser
;           l'effet de scrolling
;
cnty        LD    hl,#4000
            LD    a,h
            ADD   a,8
            LD    h,a
            AND   #38
            JP    nz,noca
            LD    a,h
            SUB   #40
            LD    h,a
            LD    a,l
            ADD   a,r1*2
            LD    l,a
            JP    nc,noca
            INC   h
            LD    a,h
            AND   7
            JP    nz,noca
            LD    a,h
            SUB   8
            LD    h,a
noca
            LD    (cnty+1),hl
;
;           on affiche une ligne de
;           chacun des blocs du
;           caractere grace a la pile
;           qui pointe sur le buffer
;           decrit plus haut
;
            EX    de,hl
            DI
            LD    (stack+1),sp
            LD    sp,buffer
            LD    a,8
mlp
            POP   hl
            LDI

            LDI
            LDI
            LDI
            ---
```

```
        LDI
        LDI
        LDI
        LDI
        PUSH hl
        POP  hl
        DEC  a
        JP   nz,mlp
stack   LD   sp,0
;
;       Test de la barre espace
;
scan
        LD   BC,#F40E
        OUT  (C),C
        LD   BC,#F6C0
        OUT  (C),C
        XOR  a
        OUT  (c),a
        LD   BC,#F792
        OUT  (C),C
        DEC  b
        LD   C,#45
        OUT  (C),C
        LD   B,#F4
        IN   A,(C)
        LD   BC,#F782
        OUT  (C),C
        DEC  b
        LD   C,#00
        OUT  (C),C
        EI
        AND  #80
        JP   NZ,BCLP
;
;       Fin du programme:
;       on restaure les registres
;       du CRTC,les banques de
;       memoire et le systeme
;
sys
        DI
        LD   bc,#7fc4
        LD   de,#014b
        OUT  (c),c
        OUT  (c),d
        OUT  (c),e
        LD   BC,#BC04
        OUT  (C),C
        INC  B
        LD   C,38
        OUT  (C),C
        LD   BC,#BC02
        OUT  (C),C
        INC  B
        LD   C,46
        OUT  (C),C
        LD   BC,#BC01
        OUT  (C),C
        INC  B
        LD   C,40
        OUT  (C),C
        LD   BC,#BC07
        OUT  (C),C
        INC  B
        LD   C,30
        OUT  (C),C
        LD   bc,#bc06
        OUT  (c),c
        INC  b
        LD   c,25
        OUT  (c),c
syst    LD   HL,#0000
        LD   (#38),hl
        EI
        RET
;
;       table des couleurs
;       du raster du haut
;
raster1
        DEFB #44,#44
        DEFB #55
        DEFB #57
        DEFB #5f,#5f
        DEFB #53,#4b,#5b
        DEFB #4b
;
;       table des couleurs
;       du raster du bas
;
raster2
        DEFB #43,#4b,#43
        DEFB #4a,#4a
        DEFB #4e
        DEFB #4c
        DEFB #5c,#5c
;
;       Table des couleurs
;       des encres
;
        DEFB #54,#4b,#44,#55
        DEFB #57,#5f,#53,#54
        DEFB #4b,#43,#4a,#4e
        DEFB #4c,#5c
;
;       Message du bas de l'ecran
;
mess
        DEFM Logon System
        DEFB 0
;
;       Message scrollant
;
text    DEFM 100/
        DEFB 0
;
;       Matrices des blocs
;
tabchr
v       EQU  128
        DEFB v,v,v,v
        DEFB v,v,v,v
        DEFB v,v,v,v
        DEFB v,v,v,v
p       EQU  143
        DEFB p,p,p,p
        DEFB p,p,p,p
        DEFB p,p,p,p
        DEFB p,p,p,p
q       EQU  212
        DEFB p,p,p,q
        DEFB p,p,q,v
        DEFB p,q,v,v
        DEFB q,v,v,v
s       EQU  213
        DEFB s,p,p,p
        DEFB v,s,p,p
        DEFB v,v,s,p
        DEFB v,v,v,s
t       EQU  214
        DEFB v,v,v,t
        DEFB v,v,t,p
        DEFB v,t,p,p
        DEFB t,p,p,p
u       EQU  215
        DEFB u,v,v,v
        DEFB p,u,v,v
        DEFB p,p,u,v
        DEFB p,p,p,u
;
;       Matrices des caracteres
;
tabchar
;%      (Attention,on prend en
;       fait le code ASCII du
;       caractere "/"-slash-)
;
        DEFB 0,4,1,5,0,0,4,0
        DEFB 0,1,0,1,0,4,2,0
        DEFB 0,3,1,2,4,2,0,0
        DEFB 0,0,0,4,2,0,rend en
;       fait le code ASCII du
;       caractere "/"-slash-)
;
        DEFB 0,4,1,5,0,0,4,0
        DEFB 0,1,0,1,0,4,2,0
        DEFB 0,3,1,2,4,2,0,0
        DEFB 0,0,0,4,2,0,0,0

        DEFB 0,0,4,2,4,1,5,0
        DEFB 0,4,2,0,1,0,1,0
        DEFB 0,2,0,0,3,1,2,0
        DEFB 0,0,0,0,0,0,0,0
;0
        DEFB 0,4,1,1,1,1,5,0
        DEFB 0,1,1,0,0,1,1,0
        DEFB 0,1,1,0,4,1,1,0
        DEFB 0,1,1,4,2,1,1,0
        DEFB 0,1,1,2,0,1,1,0
        DEFB 0,1,1,0,0,1,1,0
        DEFB 0,3,1,1,1,1,2,0
        DEFB 0,0,0,0,0,0,0,0
;1
        DEFB 0,0,4,1,1,0,0,0
        DEFB 0,0,1,1,1,0,0,0
        DEFB 0,0,0,1,1,0,0,0
        DEFB 0,0,0,1,1,0,0,0
        DEFB 0,0,0,1,1,0,0,0
        DEFB 0,0,0,1,1,0,0,0
        DEFB 0,1,1,1,1,1,1,0
        DEFB 0,0,0,0,0,0,0,0
;espace
;
;       Buffer pour les blocs
;
matrix  DEFS 64*24
;
;       buffer pour l'affichage
;       des lignes
;
buffer  DEFS 2*8,0
end
```

QUAND ELMAR FAIT DES SIENNES

Elmar Krieger est l'auteur de plusieurs jeux récents sur CPC. Prehistorik 2 et Super Cauldron. Il est aussi l'auteur de Zap'T'Balls « Édition avancée », qui utilise au mieux toutes les techniques de sprites possibles.

Dans ce numéro d'*Amstrad Cent Pour Cent*, nous vous proposons, en exclusivité, les routines de ce grand programmeur qui, pour vous faire plaisir, a préparé quelques RSX de derrière les fagots. Après cela, si vous n'êtes pas capable de créer des petits jeux sympa, c'est que vous n'êtes pas digne de la confiance que nous avons mise en vous.

MODE D'EMPLOI

Quatre programmes vous sont proposés. Les trois premiers doivent être encodés, puis lancés pour générer trois fichiers. Seuls les sprites et les routines en langage machine seront utilisés (l'image écran de 17 Ko pourra être supprimée par la suite).
Pour bien comprendre le principe de ces RSX, encodez le quatrième et dernier programme, et lancez-le tel quel. Vous comprendrez dès lors ce que veut dire une animation fluide sous Basic. C'est aussi ça la classe de nos lecteurs !
Pour les plus mordus d'entre vous, nous allons laisser la parole à notre ami qui va, sous forme de minicours, vous expliquer le principe de ses routines. Cela en vaut la peine.

LAISSONS-LUI LA PAROLE

Pendant les neuf ans que nous avons passés avec notre cher CPC, d'innombrables articles ont été écrits au sujet des sprites. En 1993, il est donc devenu assez difficile de trouver quelque chose dont on n'a pas encore parlé. Cherchons chez nos ancêtres, chez Ghosts'n'Goblins, par exemple, nous trouvons : « *la simulation avec deux bitplanes* ».
Une technique intéressante, même un peu géniale, qu'on utilise normalement sur PC et qui peut donner de bons résultats sur nos CPC.
Pour gagner de la vitesse, on va presser l'écran à une largeur de 64 octets en X, comme cela se fait dans 80 % des jeux sur CPC. La raison ? En assembleur, on peut utiliser l'instruction INC L (qui est plus rapide que INC HL) pour augmenter une adresse dans la Ram-vidéo. Le deuxième truc : nous nous servons des tableaux pour faire des calculs complexes. Comme nos routines vont accepter les coordonnées d'un sprite en X (0 - 63) et en Y (0 - 255), on doit les transférer en une adresse vidéo avant. Par exemple, avec cette formule :

ADR = &C000 + (Y\8)*64 + (Y MOD 8)*&800 + X

En assembleur, on utilise un tableau de 512 octets à &A400, contenant une adresse vidéo pour chaque ligne en Y. Si le registre HL contient les coordonnées X et Y, la formule se réduit à :

```
LD A,H
LD H,&A4
ADD A,(HL)
INC H
LD H,(HL)
LD L,A
```

Passons maintenant à la « simulation deux bitplanes » elle-même. Dans un mode à 16 couleurs (comme le MODE 0), quatre bits sont nécessaires pour choisir une des 16 couleurs. Le mot « bitplane » exprime que ces quatre bits ne se trouvent pas dans le même octet (comme sur CPC), mais sont partagés en quatre « bitplanes ». L'idée centrale de notre technique : si on met l'arrière-plan dans les bitplanes 0 et 1 (deux bits d'information, alors quatre couleurs comme le MODE 1) et les sprites dans les bitplanes 2 et 3 (encore quatre couleurs), on a réussi. On peut faire n'importe quoi avec les sprites dans 2 et 3 sans toucher le fond dans 0 et 1. Le seul point faible : on doit sacrifier une des quatre couleurs des sprites pour laisser le fond transparent. En MODE 0, il reste donc seulement sept couleurs. Mais honnêtement, l'aviez-vous remarqué dans Ghouls'n'Ghosts, Wonderboy ou Mission Genocide ?

D'autre part, il y a assez de points forts :
- on ne doit pas s'occuper de sauvegarder l'arrière-plan ;
- il faut seulement 50 % de mémoire ;
- en changeant les couleurs, on change aussi la priorité des sprites.

EN AVANT, MARCHE !

Le programme n° 1 génère cinq sprites de démonstration. Si vous voulez dessiner des sprites vous-même, faites ça en MODE 0 avec les quatre premières couleurs de la palette (pour les sprites ET le fond !). Puis, il faut mettre les sprites en mémoire dans une forme appropriée.

UNE OFFRE EXCEPTIONNELLE

Nous vous signalions qu'Elmar était l'auteur du fameux Zap'T'Balls. Si vous désirez recevoir la démo jouable de ce jeu (10 tableaux !), envoyez une *disquette et une enveloppe auto-adressée à :*
Olivier Martinerie, 9, avenue des Tilleuls, 92290 Châtenay-Malabry.
Le jeu complet, quant à lui, est au prix de 169 F, ou 189 F avec un T-shirt (large ou extra-large). Règlement par chèque à la même adresse. Notez enfin que le jeu fonctionne sur CPC 6128 ou 464 avec extension 64 Ko. Pour le 6128+, le mode deux joueurs n'est pas disponible. Quoi qu'il en soit, si vous désirez avoir plus d'informations, appelez le correspondant d'Elmar en France (Olivier Martinerie) au (1) 46-30-54-54 ; Fax : (1) 46-32-33-68 ; *ou écrivez-lui (voir l'adresse plus haut).*

LISTING

C'est fait par le programme n° 2. Regardez ici pour obtenir toutes les informations. La taille des sprites est mise dans un tableau, on ne doit plus s'en occuper après. Dans les lignes DATA, vous devez mettre, pour chaque sprite qui se trouve sur l'écran que vous avez dessiné, son numéro (0 - 127), sa position dans la Ram-vidéo (&C000-&FFFF), sa largeur (un multiple de quatre) et sa hauteur (en octets). Pour les sprites, j'ai réservé l'espace mémoire de &6000 à &97FF, c'est sans doute assez. Les données sont sauvegardées dans le fichier SPRITES.DAT.

Le programme n° 4 génère toutes les routines en assembleur et les tableaux de calcul pour décompacter et afficher les données.

LES COMMANDES RSX

Voilà une explication des commandes RSX (notez que pour les claviers AZERTY vous devez remplacer le « I » qui précède un RSX par « ù »). Chargez 'SPRMCODE. BIN' à l'adresse &9800, et initialisez avec un CALL &9800. Les instructions sont prêtes.

IINIT : initialise l'écran (32*32 caractères).

IPUTBKG,A,X,Y : imprime sprite n° A à la position X,Y dans les couleurs du fond (PUT BACKGROUND).

IPUTSPR,A,R,X,Y : imprime sprite n° A à la position X,Y en direction R [0 = normal, 1 = transparent] (PUT SPRITE).

IPUTSWF,A,R,X,Y : comme l'instruction IPUTSPR, mais attend un Frame Fly avant (PUT SPRITE AND WAIT FOR FRAME FLY).

IREMSPR,A,X,Y : enlève sprite n° A (REMOVE SPRITE).

Autour de chaque sprite, un cadre d'un octet en X et de quatre octets en Y est effacé. Pour simuler un mouvement, changez les coordonnées et imprimez le sprite encore une fois. Si le mouvement est plus grand que le cadre autour, des parties du premier sprite restent visibles.

Maintenant, les couleurs. Il y a quatre couleurs au fond. On les appelle COLFON0 à COLFON3. Quant aux trois couleurs des sprites, on les nomme COLSPR1 à COLSPR3. Chaque variable contient une valeur de 0 à 26. On regarde maintenant le cas où les sprites se trouvent devant (la deuxième possibilité doit être trouvée par vous-même, la ligne 360 de la démo peut vous aider...).

Donc, si les bits 2 et 3 sont à zéro, il n'y a pas de sprite et on peut voir la couleur du fond choisie par les bits 1 et 0. En Basic, il faut écrire :

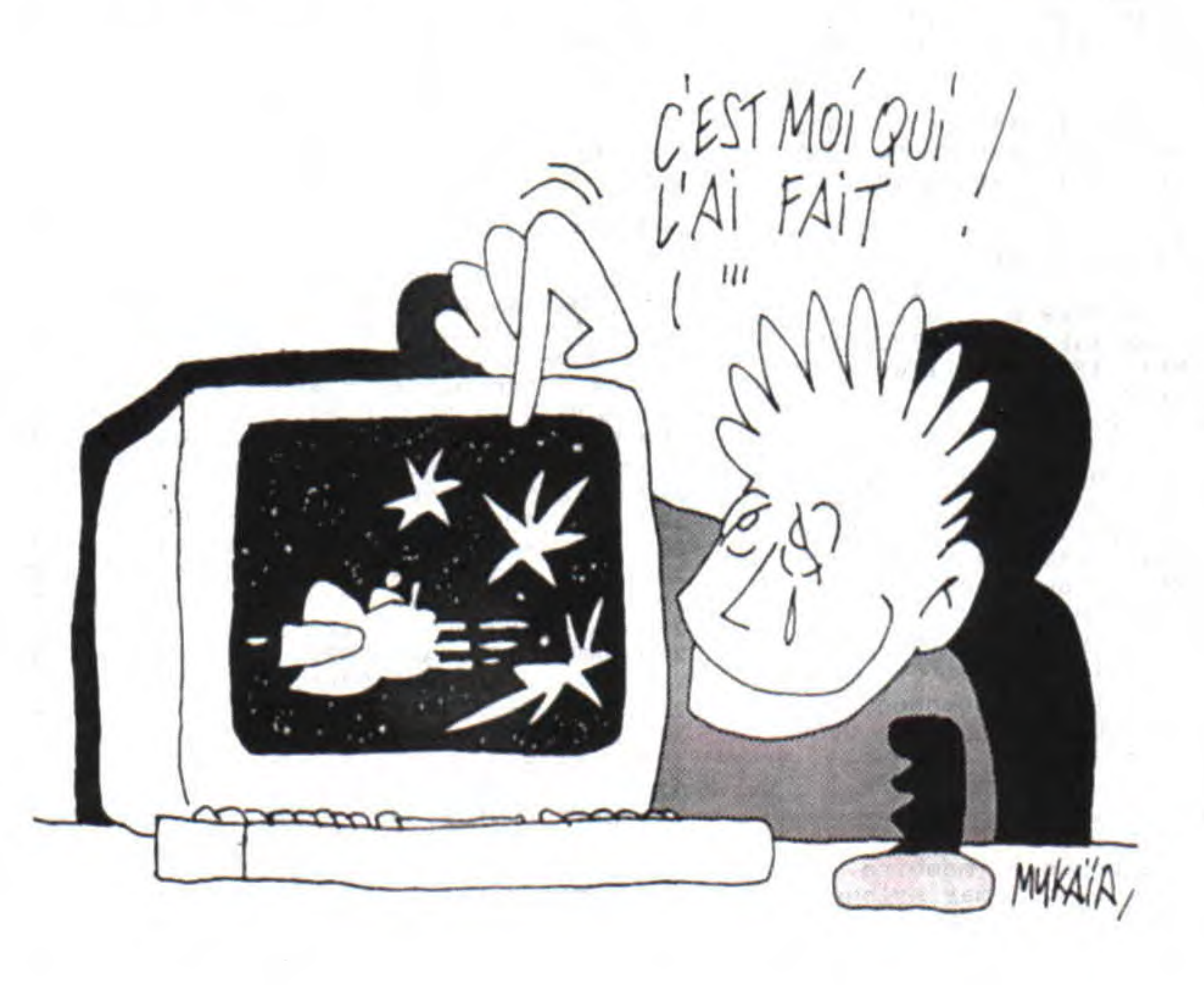
```
INK &X0000,COLFON0
INK &X0001,COLFON1
INK &X0010,COLFON2
INK &X0011,COLFON3
```

Si les bits 2 et 3 ne sont pas vides, la couleur du fond est supprimée :

```
INK &X01YY,COLSPR1
(&X0100,&X0101,&X0110,&X0111)
INK &X10YY,COLSPR2
(&X1000,&X1001,&X1010,&X1011)
INK &X11YY,COLSPR3
(&X1100,&X1101,&X1110,&X1111)
```

Une démonstration de toutes les instructions se trouve dans le programme 4.

Elmar Krieger (Elmsoft)



```
100 'Creation des sprites pour la demo.
110 '(C) AMSTRAD 100% & Elmar Krieger
120 MODE 0:INK 0,0:INK 1,13:INK 2,9:INK
3,3
130 FOR a=0 TO 3:READ adr,br,ho:FOR b=1
TO ho:FOR c=0 TO br-4 STEP 4:READ a$:d=V
AL("&"+LEFT$(a$,2)):POKE adr+c,d AND &CC
:POKE adr+c+1,(d AND &33)*4:d=VAL("&"+RI
GHT$(a$,2)):POKE adr+c+2,d AND &CC:POKE
adr+c+3,(d AND &33)*4:NEXT
140 adr=adr+&800-&C050*(adr>&F7FF):NEXT:
NEXT
150 'Fantome
160 RAD:FOR a=0 TO PI STEP 0.04:b=320+60
*COS(a):MOVE b,296+100*SIN(a):DRAW b,296
,1:DRAW b,182-20*SIN(a*8+PI),2:MOVE b,29
6:DRAW b,196-20*SIN(a*8),1:DRAWR 0,-16,3
:NEXT
170 FOR a=0 TO PI STEP 0.08:b=12*COS(a):
c=320+40*SIN(a):d=SIN(a)*80:MOVE 300+b,c
:DRAWR 0,-d,0:MOVE 336+b,c:DRAWR 0,-d:NE
XT:LOCATE 10,3:PRINT CHR$(22);CHR$(1);:P
EN 1:FOR a=0 TO 5:PRINT CHR$(207);CHR$(2
07);CHR$(8);CHR$(8);CHR$(10);:NEXT
180 SAVE "test.scr",b,&C000,&4000:END
190 'Cloture et terre
200 DATA &C000,4,48
210 DATA 2000,6020,6020,6060,E060,C0F0
220 DATA F0F0,F000,0060,48E0,4868,0C48
230 DATA 850C,8585,0F0F,0F0F,200A,4185
240 DATA 4B96,5043,964A,2C5A,4B5A,5B4F
250 DATA 5F0F,2E2F,2FBD,9FBF,BF2D,4A1B
260 DATA 5F6D,266B,73EE,76D9,73EE,E6D9
270 DATA DDE6,E6DD,D5F7,A2F6,73EA,F6D9
280 DATA 73EE,E6D9,D9E6,E6DD,DDF7,AAFE
290 'Croix
300 DATA &C008,8,9
310 DATA 0010,8000,0010,8800,0010,8000
320 DATA 0010,8800,10F0,F080,10F0,F088
330 DATA 10F0,F080,0010,8800,0010,8000
340 'Pierre tombale
350 DATA &C010,8,16
360 DATA 10F0,F080,30FA,FAC8,31F5,F5CC
370 DATA 31F8,F0C8,31F2,F1CC,31FA,F4C8
380 DATA 31F2,F9CC,31F8,F0C8,31F0,F1C4
390 DATA 30F8,D3C8,35F2,97C4,25BC,B5CE
400 DATA 07B7,BF4E,1EAD,AF8E,1F0F,2D8F
410 DATA 0F0F,0F0F
420 'Chevalier (sur balai?)
430 DATA &C018,8,31
440 DATA 0000,0C00,0001,C200,0012,E000
450 DATA 0012,0E00,0025,C000,0016,E000
460 DATA 0016,4A00,0034,E000,0032,E000
470 DATA 0032,EE00,013D,EF00,01B5,FE00
480 DATA 12F0,FE00,03B4,F400,101E,E100
490 DATA 1087,4B00,10B4,E000,0096,F000
500 DATA 0069,B400,0069,EE00,6625,E000
510 DATA F70F,E108,11EF,5B8C,FAEB,B68C
520 DATA BB1E,8708,6601,4800,0000,C000
530 DATA 0001,C000,0021,E000,0021,F000
540 DATA 0030,0C00
```

———o———

```
100 'Mettre les donnees en memoire.
110 '(C) AMSTRAD 100% & Elmar Krieger
120 MEMORY &5FFF:dat=&6000:MODE 0:FOR a=
0 TO 3:READ b:INK a,b:NEXT:FOR a=4 TO 15
:INK a,0,26:NEXT:READ a$:LOAD a$,&C000
130 READ a:IF a=-1 THEN 180
140 READ quell,br,ho
150 IF br MOD 4<>0 THEN br=br+1:GOTO 150
160 br=br-2:POKE &9600+a,dat MOD 256:POK
E &9680+a,dat\256:POKE &9700+a,ho:POKE &
9780+a,br+2
170 FOR b=1 TO ho:FOR c=0 TO br STEP 2:d
=PEEK(quell+c):e=PEEK(quell+c+1):f=d OR
((e\4) AND &33):POKE dat,f:dat=dat+1:NEX
T:POKE quell,255:quell=quell+&800-&C050*
(quell>&F7FF):NEXT:GOTO 130
180 SAVE "sprites.dat",b,&6000,&3800:END
190 'Modifier pour d'autres graphismes..
200 'Couleurs du fond
210 DATA 0,13,9,3
220 'Nom de l'ecran
230 DATA test.scr
240 'Sprite num,adr ecran,format X/Y
250 DATA 0,&C000,4,48
260 DATA 1,&C008,8,9
270 DATA 2,&C010,8,16
280 DATA 3,&C018,8,31
290 DATA 4,&C020,16,120
300 DATA -1
```

———o———

```
100 'Programme generant le 'MCODE' etc.
110 '(C) AMSTRAD 100% & Elmar Krieger
120 MEMORY &97FF:MODE 1:PRINT"Attendez 2
6 secondes.."
130 RANDOMIZE 0:FOR a=&9800 TO &9B17 STE
P 12:e=0:FOR b=0 TO 5:READ a$:c=VAL("&"+
LEFT$(a$,2)):d=VAL("&"+RIGHT$(a$,2)):POK
E a+b*2,c:POKE a+b*2+1,d:e=(e+c*RND*50+d
*RND*50) AND &F:NEXT:READ e$:IF e<>VAL("
&"+e$) THEN PRINT "Erreur en ligne:";(a-
&9800)\12+19:EN
140 NEXT
150 FOR a=0 TO 255:POKE &A000+a,(a\4) AN
D &33:POKE &A100+a,a AND &33:POKE &A200+
a,((a AND &40)\2) OR ((a AND &80)\8) OR
((a AND &4)\2) OR ((a AND &8)\8):POKE &A
300+a,((a AND &10)*2) OR ((a AND &20)\2)
 OR ((a AND &1)*2) OR ((a AND &2)\2):NEX
T
160 adr=&C000:FOR a=0 TO 255:ad$=HEX$(ad
r,4):POKE &A400+a,VAL("&"+RIGHT$(ad$,2))
:POKE &A500+a,VAL("&"+LEFT$(ad$,2)):adr=
adr+&800-&C040*(adr>&F7FF):NEXT
170 SAVE "sprmcode",B,&9800,&E00
180 DATA 2109,9801,0D98,C3D1,BC00,0000,5
190 DATA 001E,98C3,3B98,C35C,98C3,5398,7
200 DATA C3C5,9AC3,789A,494E,49D4,5055,6
210 DATA 5453,50D2,5055,5453,57C6,5245,9
220 DATA 4D53,50D2,5055,5442,4BC7,0021,6
230 DATA 4C98,0E07,06BC,ED49,06BE,EDA3,7
240 DATA 0D20,F5C9,2220,0026,8E2A,204F,E
250 DATA 06F5,ED78,1F30,F979,3277,9ADD,A
260 DATA 6E00,DD66,0225,FA6A,9A7D,D604,0
270 DATA DA6A,9A6F,7C26,A486,2466,6FEB,0
280 DATA DD6E,0626,974E,CBFD,46FD,6978,6
290 DATA C602,32C8,9832,7299,32A2,9932,2
300 DATA 4C9A,3E10,904F,8767,8784,8132,0
310 DATA E698,32C0,9926,967E,CBBD,6E67,9
320 DATA ED73,689A,F3F9,DD7E,043D,CA8B,2
330 DATA 993E,FF90,32D1,9832,5C99,327B,3
340 DATA 990E,CCFD,2604,EB06,007E,A177,4
350 DATA 2C10,FA2D,1100,0719,3004,1140,4
360 DATA C019,FD25,20E9,EB0E,CC1A,A112,6
370 DATA 1C18,00E1,4426,A01A,A1B6,1224,5
380 DATA 1C1A,A1B6,1225,1C68,1AA1,B612,4
390 DATA 241C,1AA1,B612,1CE1,4426,A01A,8
400 DATA A1B6,1224,1C1A,A1B6,1225,1C68,6
410 DATA 1AA1,B612,241C,1AA1,B612,1CE1,9
420 DATA 4426,A01A,A1B6,1224,1C1A,A1B6,7
430 DATA 1225,1C68,1AA1,B612,241C,1AA1,8
440 DATA B612,1CE1,4426,A01A,A1B6,1224,2
```

```
450 DATA 1C1A,A1B6,1225,1C68,1AA1,B612,F
460 DATA 241C,1AA1,B612,1C1A,A112,EB01,5
470 DATA 0007,0930,0401,40C0,09EB,FD2D,7
480 DATA C2DF,980E,CCFD,2604,EB06,007E,7
490 DATA A177,2C10,FA2D,1100,0719,3004,8
500 DATA 1140,C019,FD25,20E9,C362,9A3E,1
510 DATA 0180,32AB,9932,369A,3255,9A7B,6
520 DATA 805F,1C0E,CCFD,2604,EB06,007E,5
530 DATA A177,2D10,FA2C,1100,0819,3004,B
540 DATA 1140,C019,FD25,20E9,EB0E,CC1A,0
550 DATA A112,1D18,00E1,4426,A21A,A1B6,B
560 DATA 1224,1D1A,A1B6,1225,1D68,1AA1,0
570 DATA B612,241D,1AA1,B612,1DE1,4426,A
580 DATA A21A,A1B6,1224,1D1A,A1B6,1225,F
590 DATA 1D68,1AA1,B612,241D,1AA1,B612,B
600 DATA 1DE1,4426,A21A,A1B6,1224,1D1A,0
610 DATA A1B6,1225,1D68,1AA1,B612,241D,D
620 DATA 1AA1,B612,1DE1,4426,A21A,A1B6,7
630 DATA 1224,1D1A,A1B6,1225,1D68,1AA1,B
640 DATA B612,241D,1AA1,B612,1D1A,A112,C
650 DATA EB01,0008,0930,0401,40C0,09EB,7
660 DATA FD2D,C2B9,990E,CCFD,2604,EB06,0
670 DATA 007E,A177,2D10,FA2C,1100,0819,0
680 DATA 3004,1140,C019,FD25,20E9,1108,8
690 DATA 00DD,1931,0000,FB3A,779A,D604,4
700 DATA 3277,9AC2,5F98,C900,ED47,CDF2,1
710 DATA 9ACB,3AFD,6A4B,EBDD,6E04,2696,B
720 DATA 7ECB,FD66,6FEB,FD45,7DFD,671A,2
730 DATA E6CC,772C,1A17,17E6,CC77,2C13,F
740 DATA 10F1,FD7C,6F7C,C608,6730,083E,6
750 DATA 4085,6F3E,C08C,670D,20D8,1106,4
760 DATA 00DD,19ED,57D6,03ED,47C2,7A9A,2
770 DATA C9FD,67CD,F29A,425D,0ECC,7EA1,6
780 DATA 772C,10FA,6B7C,C608,6730,0401,B
790 DATA 40C0,09FD,2D20,E711,0600,DD19,B
800 DATA FD7C,D603,FD67,C2C7,9AC9,DD5E,9
810 DATA 00DD,5602,DD6E,0426,977E,CBFD,1
820 DATA 666F,EBFD,6B7C,26A4,8624,666F,E
830 DATA C900,0000,0000,0000,0000,0000,C
```

———————— 0 ————————

```
100 'Demo des commandes RSX
110 '(C) AMSTRAD 100% & Elmar Krieger
120 MEMORY &5FFF:MODE 0:LOAD "sprites.da
t",&6000:LOAD "sprmcode",&9800
130 'Initialiser
140 CALL &9800:|INIT:MODE 0:hgfa(1)=0:hg
fa(2)=13:hgfa(3)=9:hgfa(4)=3:spfa(1)=26:
spfa(2)=2:spfa(3)=15:v=1:GOSUB 310
150 LOCATE 1,1:PRINT"UN CIMETIERE....":F
OR a=0 TO 60 STEP 4:|PUTBKG,0,a,208:NEXT
:b=0:FOR a=6 TO 48 STEP 14:x(b)=a:y(b)=1
99:b=b+1:|PUTBKG,1,a,199:|PUTBKG,2,a,208
:NEXT:h(0)=85:h(1)=40:h(2)=60:h(3)=75:GO
SUB 300
160 LOCATE 1,1:PRINT"MAL S'APPROCHE..":F
OR a=0 TO 3:FOR b=0 TO h(a):CALL &BD19:|
PUTBKG,1,x(a),y(a):y(a)=y(a)-2:NEXT:NEXT
:GOSUB 300
170 LOCATE 1,1:PRINT"LE GRAND SAUVEUR":x
=28:y=177:r=0:FOR a=0 TO y:|PUTSWF,3,r,x
,a:NEXT
180 LOCATE 1,1:PRINT"CURS.,COPY,SPACE"
190 |PUTSWF,3,r,x,y:x2=x
200 x=x-INKEY(8)+INKEY(1)+(x>53)-(x<2):y
=y-4*INKEY(0)+4*INKEY(2)+4*(y>193)-4*(y<
8):IF x2=x THEN 210 ELSE IF x2<x THEN r=
0 ELSE r=1
210 IF NOT INKEY(9) THEN GOSUB 310
220 IF INKEY(47) THEN 190
230 |REMSPR,3,x,y
240 LOCATE 1,1:PRINT"  IL VIENT....  ":|
PUTSPR,3,0,2,191:GOSUB 300
250 v=0:r=0:GOSUB 310:x=24:GOSUB 290:WHI
LE INKEY$="":FOR a=130 TO 20 STEP -4:|PU
TSPR,4,r,x,a:NEXT:FOR a=0 TO 10:|PUTSPR,
4,r,x,20:r=r XOR 1:FOR b=1 TO 200:NEXT:N
EXT:GOSUB 310:FOR a=20 TO 130 STEP 4:|PU
TSPR,4,r,x,a:NEXT:WEND
260 v=0:GOSUB 310:LOCATE 1,1:PRINT "C'ES
T COURAGEUX!":GOSUB 300
270 FOR a=191 TO 6 STEP -1:|PUTSWF,3,0,2
,a:NEXT:FOR a=2 TO 28:|PUTSWF,3,0,a,6:NE
XT:FOR A=6 TO 94 STEP 4:|PUTSWF,3,0,28,a
:NEXT:FOR a=130 TO 250 STEP 4:POKE &9704
,251-a:|PUTSWF,3,0,28,a-36,4,0,x,a:NEXT:
FOR A=214 TO 6 STEP -1:|PUTSWF,3,0,28,a:
NEXT
280 LOCATE 1,1:PRINT" LA VICTOIRE ?  ":G
OSUB 300:LOCATE 1,1:PRINT "AAAARRRRGGHH!
!!!":OUT &BC00,8:OUT &BD00,1:x=4:GOSUB 2
90:x=24:GOSUB 290:x=44:GOSUB 290:OUT &BD
00,0:LOCATE 1,1:PRINT "NO HAPPY END...."
:GOSUB 300:GOTO 370
290 FOR a=250 TO 130 STEP -4:POKE &9704,
251-a:|PUTSWF,4,0,x,a:NEXT:RETURN
300 WHILE INKEY$<>"":WEND:CALL &BB18:RET
URN
310 'Couleurs
320 v=v XOR 1:IF v=1 THEN 350
330 'Sprites devant
340 FOR a=0 TO 3:INK a,hgfa(a+1):NEXT:FO
R a=4 TO 12 STEP 4:FOR b=a TO a+3:INK b,
spfa(a/4):NEXT:NEXT:RETURN
350 'Sprites derriere
360 FOR a=1 TO 3:FOR b=1 TO 3:INK a*4+b,
hgfa(b+1):NEXT:NEXT:FOR a=4 TO 12 STEP 4
:INK a,spfa(a/4):NEXT:RETURN
370 'Fin (reset CRTC)
380 OUT &BC00,1:OUT &BD00,40
390 OUT &BC00,2:OUT &BD00,46
400 OUT &BC00,6:OUT &BD00,25
410 OUT &BC00,7:OUT &BD00,30
420 END
```

HÉBÉTÉ ET LOBOTOMISÉ PRIMAIRE

En effet, je me suis toujours demandé pourquoi on rejette les plus petits. Enfin, cela ne fait rien. On va continuer la mégarécap' de toutes les solutions parues dans *Amstrad Cent Pour Cent*.

Je signale à ceux qui n'ont pas suivi tous les épisodes, que cette récap' a commencé dans le dernier numéro. Pour le commander, c'est très simple. Il vous suffit de téléphoner au *(16)* 44-72-77-55 (c'est en province) pour avoir tous les renseignements possibles et imaginables. Je tiens à vous signaler aussi qu'il ne faut pas écouter Totov. En effet, en ayant décortiqué son dernier article sur les fanzines, vous remarquerez qu'il dit, dans son langage approximatif, que je ne sais pas faire des vu-mètres. C'est archi-faux, on se demanderait presque où il a pu chercher pareilles bêtises si on ne connaissait pas ses antécédents.

LE RÉCAPITULATIF

On en était resté au *numéro 35*.

Saga de Lankhor : un magnifique plan qui mérite le coup d'œil.

L'Ile d'Ubi Soft : la solution complète de ce jeu qui je le rappelle a été programmé par Poum lui-même en personne et dont les graphismes ont été réalisés par Laurent Théron.

Mokowe de Lankhor : quelques trucs pour ce jeu malheureusement trop dur et surtout pas très ergonomique.

Passons au *numéro 36*, maintenant.

Daw de Lankhor : la solution complète de ce jeu qui causa bien des problèmes à notre rédac'chef.

Saga de Lankhor : la solution de l'un des seuls jeux qui possèdent des graphismes en mode 2, on peut de plus jouer à deux, ce qui est fort recommandé.

Chip's Challenge d'US Gold : quelques codes en vrac.

Comme il n'y avait pas de rubrique Help dans le *numéro 37* passons au *numéro 38*.

Mokowe de Lankhor : la solution de ce jeu qui, comme je le disais plus haut, n'était pas très ergonomique, m'enfin, c'est quand même un classique du jeu d'aventure.

Numéro 39

Sphaira d'Ubi Soft : après de nombreuses suppliques de Poum, nous reçûmes enfin la solution de ce jeu assez vieux (il fut testé, je crois, dans l'un des tout premiers numéros d'*Amstrad Cent Pour Cent*).

Sim City : une astuce du tonnerre pour ce jeu génial.

Bat : considéré comme le meilleur jeu d'aventure par la plupart d'entre nous. Il y avait un petit cheat mode qui permettait de posséder tous les objets du jeu. Le voici : dans la salle d'arcade, devant les machines, activez le menu principal, sélectionnez l'option « recherche » ou « examiner » et vous verrez...

Numéro 40

Batman, The Movie : le plan du premier niveau de ce jeu qui commence à dater.

Fugitif : la solution de ce jeu géant. Je vous signale aussi que de nombreuses pétouilles s'y étaient glissées et qu'elles ont été corrigées dans le *numéro 41* (dans les Actus).

Numéro 41

Orphée : la solution du jeu fétiche de Poum.

Bat : voici la solution de ce merveilleux soft qui a tenu en haleine des tas et des tas de personnes.

Numéro 42

Infernal House de Lankhor : la solution complète de ce jeu qui est, sans aucun doute, la meilleure production de chez Lankohr.

MGT de Loriciel : la solution de cet excellent jeu, mais très peu connu.

Robocop 2 : le plan du premier level du meilleur jeu en cartouche conçu sur GX-4000 et CPC+.

Numéro 43

Alive de Lankhor : la solution complète de ce jeu très particulier.

La Crypte des Maudits de Lankhor : la solution de ce jeu qui, hélas ! marqua la fin de la production de ses auteurs. Souvenez-vous, ils avaient déjà réalisé La Secte noire (La Crypte des Maudits, c'est la suite) et Mokowe, tous deux de grands hits qui n'ont pas arrêté de nous creuser les méninges, pour avoir ne serait-ce qu'un tout petit bout de la solution. Ahhh, c'était le bon temps... De plus, je viens d'avoir la correction de la solution, par Nicolas Boullis, juste avant « pousser soufflet », il manque « ouvrir porte » et « EST » dans la seconde partie, au lieu de « gratter suite » il faut mettre « gratter suie » et enfin ce n'est pas « prendre urne et examiner urne », il faut remplacer « urne » par « rune ».

Night Shift : le code du premier niveau (tenez le voici : cerise, cerise, banane, prune).

Builderland : les quatre codes de ce jeu « prise de tête ». Les voici dans l'ordre : volcan, darchy, nebula, hyblis.

Supercars de Gremlin : les deux codes « Odie et Bigc ».

Simpsons : le plan du premier niveau de ce super jeu.

Numéro 44

Simpsons : après le plan dans le *numéro 43*, voici la soluce. On n'imagine pas le nombre de nuits blanches qu'a dû passer Sukie pour finir ce jeu démentiel.

Numéro 45

Il n'y avait pas de rubrique Help dans ce numéro, alors au suivant...

Numéro 46

C'est ce numéro qui marqua mon entrée dans ACPC, j'y présentais des tas de soluces et astuces comme :

Survivre de Lankhor : c'est Totov qui, le premier, en avait trouvé la solution, c'est normal, il l'a reçu avant les lecteurs et c'est lui qui l'avait testé.

Le Trésor D'Ali Gator de Lankhor : là c'est moi le plus rapide sur le coup, je vous en présentais la solution complète détaillée. À noter qu'il doit y avoir une autre solution probable. À vous de chercher... L'auteur de ce jeu est Claude Le Moullec, et même si son jeu n'était pas très réussi et même s'il ne produit plus de listing sur CPC, je resterais longtemps l'un de ses fans...

La Crypte des Maudits de Lankhor : une astuce fort simple qui a rencontré certains problèmes à la fabrication du journal, je l'ai donc remise dans le numéro d'après, soit le *numéro 47*.

Pepito : non, je ne passe pas du coq à l'âne, car il y a vraiment eu un jeu

qui s'appelait ainsi. Le jeu, d'ailleurs, vous permet d'incarner ce petit héros mexicain qui adore, comme Sined, les petits gâteaux ronds nappés de chocolat.

Sylva de Lankhor : la solution de ce jeu qui fut le dernier réalisé par cette société que l'on aimait bien.

Lemmings : les codes des vingt premiers niveaux de ce classique du jeu de réflexion.

Numéro 47

La Crypte des Maudits : l'astuce qui était présente dans le 46 était tronquée, il était donc normal que je la rappelle dans ce numéro.

Zap'T'Balls : les codes de l'un des meilleurs jeux jamais réalisés sur CPC, contrairement à ce que j'ai pu dire, il ne manque aucun code pour finir le jeu.

Saga : une méga-astuce qui mérite un coup d'oeil.

Et enfin, le début du récapitulatif que je termine en ce moment.

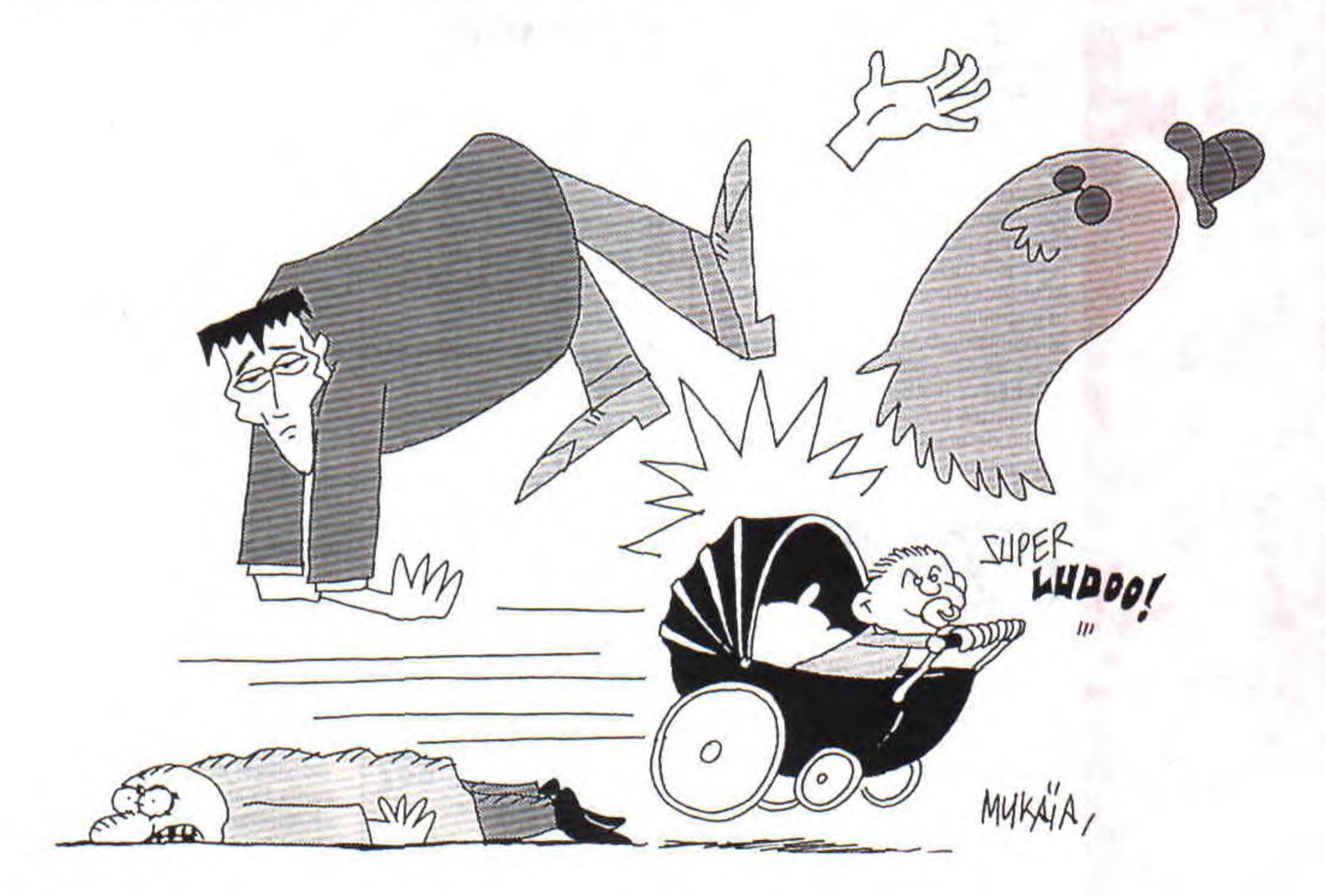
ADDAMS FAMILY

Voici maintenant la solution (en mode « easy ») du jeu basé sur la famille la plus loufoque existant sur Terre. Comme elle est longue, pas trop de bla-bla. Comme cela semble logique, il faut prendre les cœurs, points ou autres bonus, ainsi que les personnages, en cherchant la technique qui vous semble la plus appropriée (il faut que les lecteurs aussi se décarcassent et, en plus, il n'y a plus beaucoup de place). Prendre porte, aller en bas à gauche, prendre porte, gauche pendant 3 tableaux, prendre clef rouge sur l'armoire jaune, retour au point de départ, aller complètement à gauche, prendre porte, continuer à gauche, sauter sur la tête de Lurek : vous devez rester vivant pendant une minute en sautant par-dessus l'étoile blanche, prendre porte, droite, porte, gauche, prendre clé violette, retour au point de départ, porte, bas à droite, droite, porte, droite pendant 4 tableaux, porte, gauche pendant 3 tableaux, porte, bas, gauche pendant 2 tableaux, bas, droite, porte, gauche, sauter sur la chose (pour ceux qui ne le savent pas, c'est la main, c'est une amie, alors n'ayez pas peur de la toucher) : vous devez éviter le machin en vous aidant des têtes de mort, droite, porte, gauche, haut, droite pendant 2 tableaux, haut, porte, droit pendant 3 tableaux, porte du haut, bas, droite, porte, gauche pendant 3 tableaux, sauter sur la tronche de Grammy : il faut sauter d'une tête de squelette à l'autre sans tomber. Retournez à l'écran où il y avait la clé violette (dehors), porte, droite, haut + gauche, porte, droite, porte, gauche, porte, porte du bas, haut (grâce aux branches), prenez la clé bleue, retour au début, porte, porte du haut, porte du bas, porte du haut (il faut se servir du fantôme comme tremplin), porte du bas, droite pendant 2 tableaux, porte, droite pendant 5 tableaux, (monter) gauche, porte, droite, bas, gauche pendant 3 tableaux, bas, gauche, prendre la clé jaune (mettez-vous entre la 2[e] et la 3[e] étoile du bas, attendez que le poisson vienne sur vous, puis montez et jetez-vous sur la clé en évitant le piranha), retour au début, porte, haut + gauche, gauche pendant 4 tableaux, porte du haut, gauche, retour au début, gauche pendant 3 tableaux, porte, droite pendant 3 tableaux, porte, droite, bas, droite, porte de gauche, gauche, monter, droite, porte du haut, gauche (les serpents volants peuvent servir de trampoline), porte, monter, droite, prendre la clé blanche, gauche pendant 4 tableaux, droite pendant 3 tableaux, porte, droite, porte, porte du bas, monter à gauche, haut, gauche, porte, gauche (toujours avec les ennemis comme trampolines) pendant 3 tableaux, porte, gauche pendant 3 tableaux, porte, droite pendant 5 tableaux, porte du haut, porte du bas, porte du haut, droite, descendre, gauche, porte du bas, Attention ! voici le labyrinthe. NE VOUS PERDEZ PAS EN ROUTE !

Bas, porte, gauche, bas, droite pendant 6 tableaux, haut, droite, porte, droite, bas, droite, porte, droite, porte de droite, porte du haut, gauche, porte et re-porte, diagonale, bas-gauche, gauche pendant 3 tableaux, porte, droite, haut, monter, gauche, prendre chaudron + haut dans le passage du milieu, droite, monter, gauche, porte, porte de droite, gauche pendant 2 tableaux. Voilà Morticia, sautez-lui dessus, puis évitez l'ennemi sans vous enfoncer dans les sables mouvants.

Voilà, vous avez terminé et Étienne JULIOT, l'auteur de la soluce, gagne un bon d'achat de 250 F chez notre ami Jessico (le mode d'emploi est indiqué dans le dernier numéro). Alors à +++ et envoyez-moi vos soluces.

Ludotronic



VECTEUR VICTOIRE VICTOR

Encore une nouvelle rubrique, qui va vous permettre de faire plein de choses formidables sur vos écrans chéris, et dans laquelle nous allons traiter uniquement du graphisme en programmation, des déformations et autre agrandissements, ainsi que de la 3D et autres types d'algorithmes sympa. Mais attention ! ici pas d'animation. Uniquement du graphisme.

D'abord, pourquoi remplir cette rubrique de choses dont parlent uniquement les profs de maths et les agrégés de physique ? Eh bien, parce que nous allons nous attaquer au graphisme vectoriel. Ne fuyez pas, vous allez voir que ce n'est pas si compliqué que ça. En effet, sans le savoir, il vous est sûrement déjà arrivé d'utiliser ce type de graphismes si vous programmez un temps soit peu avec des DRAW.

BAH POURQUOI ?

En fait, nous n'utiliserons pas exactement les mêmes types de vecteurs qu'en mathématiques, mais une sorte de purée prédigérée qui va nous permettre de coller très facilement des effets spéciaux à nos jolis dessins.

Quand vous programmez des graphismes avec des MOVE et des DRAW, vous utilisez souvent les coordonnées directes des points sur l'écran sans manipuler de DATA, ce qui fait qu'il vous faut souvent perdre du temps à rechercher les coordonnées d'un point dans une bouillie de DRAW et de MOVE. Ce que je vous propose donc, c'est d'architecturer votre salmigondis d'instructions en plusieurs groupes distincts de DATA et deux simples boucles pour mémoriser les points et afficher les lignes, ce qui vous permettra de ne modifier qu'une partie de boucle, plutôt que tout le programme, pour créer des effets spéciaux.

ARCHITECTURONS EN CŒUR

Une ligne est constituée de deux choses, les points de départ et d'arrivée, et la ligne elle-même.

Pensez à ces jeux pour gamins, remplis de points numérotés qu'il faut relier entre eux dans un ordre bien précis et qui permettent d'obtenir un zoli dessin. Eh bien, nous allons faire pareil.

On va rassembler, dans un tableau de DATA, toutes les coordonnées des points, et dans un autre tableau, on va indiquer le numéro des points de départ et d'arrivée de chacune des lignes qui constituent le dessin, ainsi que la couleur de ces mêmes lignes. Chacun des tableaux commence par un nombre représentant le nombre d'éléments qu'il contient, ce qui permet d'éviter certains plantages.

INCONVÉNIENTS

À part le fait qu'il y a des liaisons lignes-points à manipuler, et que ce n'est pas très évident quand on n'a pas l'habitude, il n'y a pas vraiment d'inconvénients.

AVANTAGES

Cette méthode est extensible, on peut rajouter une troisième coordonnée aux points pour des projections en 3D, ajouter des attributs à chacune des lignes (couleurs, taille, inversion vidéo), créer un troisième tableau pour des formes géométriques à partir de lignes ou de cercles.

Si vous faites des jeux de rôle, les DATA peuvent être rangés dans des fichiers Ascii. Chaque fichier pouvant être lu pendant le jeu, sans empiéter dans la mémoire, vous pouvez créer des jeux ayant plus de cent salles différentes avec un minimum de programmation et une mémoire libre pour d'éventuels RSX ou données sonores.

Avec la méthode par fichier, les dessins peuvent être développés en-dehors du programme de jeu. Les DATA sont largement plus faciles à lire (s'ils sont bien ordonnés) qu'un long blabla de DRAW et de MOVE. On peut afficher le numéro de la ligne et les numéros des points de cette ligne pendant l'affi-

```
10 MODE 1:INK 0,0:INK 1,1:INK 2,26
20
30 'mémorisation des points
40 RESTORE 270
50 READ nb%
60 nbpoint% = nb% - 1
70 DIM x%(nbpoint%), y%(nbpoint%)
80 FOR n% = 0 TO nbpoint%
90 READ x%(n%), y%(n)%
100 NEXT
110
120 'lecture et affichage des lignes
130 RESTORE 340
140 READ nb%
150 nbligne% = nb% - 1
160 FOR n% = 0 TO nbligne%
170 READ d%, a%, c%
180 d% = d% - 1
190 a% = a% - 1
200 IF d% > nbpoint% OR a% > nbpoint% THEN PRINT "erreur de No de
point ligne "; n%: END
210 MOVE 300+2*x%(d%),190+2*y%(d%)
215 DRAW 300+2*x%(a%),190+2*y%(a%), c%
220 'LOCATE 1,12:PRINT "ligne ";n%:WHILE INKEY$="":WEND
230 NEXT
240 END
250
260 '4 points x,y
270 DATA 4
280 DATA   0,  0
290 DATA  10,  0
300 DATA  10, 10
310 DATA   0, 10
320
330 '6 lignes depard,arrivee
340 DATA 6
350 DATA  1, 2, 1
360 DATA  2, 3, 1
370 DATA  3, 4, 1
380 DATA  4, 1, 1
390 DATA  1, 3,15
400 DATA  2, 4,15
```

```
10 MODE 1:DEG
20 FOR a = 0 TO 45 step 0.5
30 x = COS(a) * 150
40 y = SIN(a) * 150
50 PLOT 320 + x, 200 + y
60 PLOT 320 - x, 200 + y
70 PLOT 320 - x, 200 - y
80 PLOT 320 + x, 200 - y
90 PLOT 320 + y, 200 + x
100 PLOT 320 - y, 200 + x
110 PLOT 320 - y, 200 - x
120 PLOT 320 + y, 200 - x
130 NEXT
```

chage, avec une attente de touche, ce qui permet de trouver les erreurs en un temps record lors de la conception du dessin. Il suffit de modifier la position d'un point pour que toutes les lignes qui lui sont liées soient modifiées. Bref, cette méthode est la meilleure, et on peut aussi l'appliquer sur des arcs de cercle, des courbes de Bézier, des cercle, etc.

DE VISU

Voici un exemple qui affiche un carré bleu avec une croix blanche à l'intérieur.
(Voir le listing ci-avant)

Pour ce qu'il fait ce programme peut vous paraître long, mais quand il s'agira de dessiner une salle, avec tous ses détails, pour un jeu, c'est cette méthode que vous trouverez la plus courte pour la masse de détails à gérer.

MULTIPLIONS

Avec cette méthode, il vous est aussi possible de créer des graphismes que vous pouvez afficher en plusieurs endroits à l'écran et de plusieurs tailles. Il n'y a pas vraiment de limites, à part évidemment celles de l'imagination.

LES CERCLES

Tout ça, c'est bien pour faire des dessins, mais ça serait encore mieux si on pouvait aussi dessiner des ronds. Il y a plein de façons différentes pour faire des cercles, mais la plus simple c'est l'équation x=COS(a)*r,y=SIN(a)*r, où r est le rayon et a l'angle variant de 0 à 360. Bien sûr, il faut utiliser une boucle pour faire varier a. Le problème, c'est qu'avec ce calcul, si on utilise des PLOT, ça risque de mettre un temps fou pour tracer un cercle correct, quel que soit le rayon, car les calculs avec cosinus et sinus sont très lents, vu la complexité de ce type de fonctions. On peut tracer un cercle huit fois plus rapidement, en faisant une boucle qui ne varie que de 0 à 45 degrés d'angle. Les cercles sont symétriques en vertical, en horizontal, en diagonales gauche et droite, ce qui fait quatre symétries parfaites, ce qui veut dire qu'en ne calculant que le huitième on peut tracer un cercle.
(Voir le listing ci-dessus)

ALGO BALÈZE

Ça c'est la méthode traditionnelle pour dessiner un cercle, mais elle est quand même très lente, puisque si vous regardez bien l'algorithme, vous remarquerez que le programme perd du temps à réécrire trop souvent sur le même point. Comment faire pour qu'il n'écrive qu'un seul point et ensuite qu'il passe directement au point suivant en faisant un minimum de calculs ?

RÉFLÉCHISSONS

Prenons un cercle normal dessiné avec la fonction décrite au-dessus ou un cercle dessiné avec un programme de dessin tel qu'OCP. Puisqu'il suffit de ne tracer qu'un huitième de cercle, le reste étant obtenu par symétrie, concentrons-nous uniquement sur l'arc de cercle partant de 90 degrés jusqu'à 45 degrés. Collons-nous le nez sur l'écran et regardons de plus près ce qui s'y passe : nous remarquons que les points ne se placent uniquement qu'à droite ou en bas et toujours à droite du point précédent, ce qui veut dire que le point est toujours décalé vers la droite, mais qu'il peut être sur la même ligne ou sur la ligne d'en-dessous. Qu'est-ce qui détermine le choix entre la même ligne ou la ligne d'en-dessous ? Eh bien ! c'est la distance entre le point et le centre du cercle. Exemple : (toujours dans l'arc de 90 degrés à 45 degrés) on place un point n'importe où sur cet arc de cercle, le point suivant sera sur la même ligne s'il n'est pas en dehors du cercle, sinon il est sur la ligne en-dessous.
(Voir le listing ci-dessous)

Bon, voilà, c'est terminé, dans le prochain numéro, nous étudierons les fonctions de projection 3D, alors, see you soon!

Le Grapheux !

```
10 MODE 1
20 R=150:CX=320:CY=200
30 Y=R:X=0:XSTOP=R*SQR(2)/2
40 GOSUB 110
50 WHILE X<=XSTOP
60 X=X+2
70  IF SQR(X^2+Y^2)>R+1 THEN Y=Y-2
80  GOSUB 110
90 WEND
100 END
110 PLOT CX+X,CY+Y
120 PLOT CX-X,CY+Y
130 PLOT CX+X,CY-Y
140 PLOT CX-X,CY-Y
150 PLOT CX+Y,CY+X
160 PLOT CX-Y,CY+X
170 PLOT CX+Y,CY-X
180 PLOT CX-Y,CY-X
190 RETURN
```

FER A SOUDER

COMMENT TRANSFORMER UNE GX-4000 EN CPC PLUS

Eh oui, ça devait arriver... La GX-4000 étant un sous-produit du CPC+, il paraissait concevable de greffer à la console un clavier et un drive, vous offrant ainsi à peu de frais un ordinateur doté des capacités totales d'un CPC+ (vous devriez trouver la console à environ 200 F sur le marché de l'occasion puisqu'elle n'est plus commercialisée).

Cette transformation est bien entendu possible à condition de disposer du clavier et du drive d'un ancien CPC dont vous n'avez plus l'utilité. Mais vous pouvez envisager de trouver ces éléments sur votre ancien CPC si vous désirez lui donner des ailes.

Le montage a été réalisé par un démomaker appelé « The Cranium ». Celui-ci m'avait présenté son étonnant prototype lors de l'Euro Meeting 2. Je transcris donc ici l'article que je lui avais commandé pour que le maximum de personnes puissent réaliser le CPC-GX+ ! Si vous désirez poser des questions à l'auteur à propos du montage, écrivez-lui en allemand ou en anglais à l'adresse suivante :

Wolfgang Noisternig
Friedrich-Ebert-Str. 27
D-2418 RATZEBURG
ALLEMAGNE

DE QUOI AVEZ-VOUS BESOIN ?

Voilà la liste des objets que vous devrez vous procurer pour mener à bien votre projet :

- 1 GX-4000 ;
- 1 vieux clavier de CPC (seulement le clavier, pas l'ordinateur lui-même) ;
- 1 connecteur double-face 2 x 25 broches (2,54 mm - Port d'extension) ;
- 1 connecteur double-face 2 x 17 broches (2,54 mm - Port imprimante) ;
- 2 Rams 41464-12 ;
- 1 résistance 33 ohms ;
- 4 résistances 2,2 k/ohms ;
- 3 résistances 10 k/ohms ;
- 1 cartouche basic du CPC+ (ou seulement la Rom). Dans ce cas, vous devrez retirer la vieille Rom de la cartouche de votre GX-4000 et installer à la place votre copie de la Rom du CPC+ ;
- plusieurs petits fils (environ 0,1 mm) ;
- plusieurs fils normaux utilisés en électronique.

LES DIFFÉRENTES ÉTAPES

Étape 1 : ouvrez votre GX-4000 (2 vis) et cherchez l'Asic. C'est le circuit intégré avec 160 broches (vous ne pouvez pas le louper !).

Étape 2 : nous devons connecter les 2 circuits de Ram *(voir fig. 1)*.
Implantez-les sur les 2 circuits de ram existants (IC10, IC11) et connectez toutes les broches exceptée la broche 16 (pour chaque circuit). Cette broche doit être laissée libre, car c'est la sélection de circuit pour la Ram.
Vous devez ensuite connecter les 2 broches 16 ensemble.
Sur une des 2 broches, vous devez aussi connecter la résistance 33 ohms.
L'autre côté de la résistance devant être relié à la broche 91 de l'Asic.

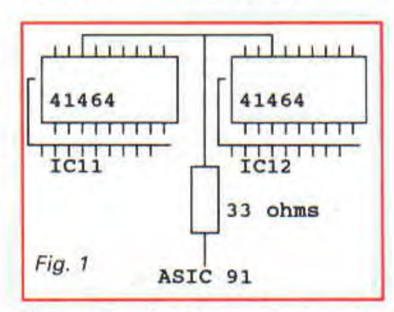


Fig. 1

Étape 3 : cherchez maintenant le circuit IC8 (c'est un 74HC4051N). Vous devez connecter les résistances 10 k/ohms aux broches 9, 10 et 11 du circuit. Les autres côtés des résistances seront reliés ensemble et connectés à la broche 8 du circuit *(voir fig. 2)*.

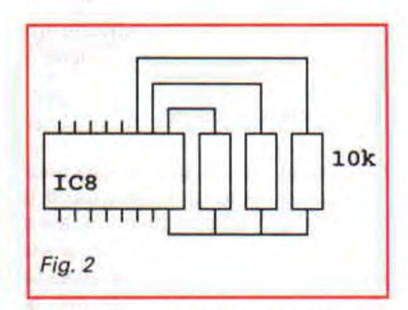


Fig. 2

Étape 4 : ensuite, connectez la broche 119 de l'Asic avec le côté du C17 qui n'est pas connecté avec l'Asic. Vous devrez connecter à cet endroit une résistance 2,2 k/ohms. L'autre côté de la résistance doit être relié à la masse *(voir fig. 3)*

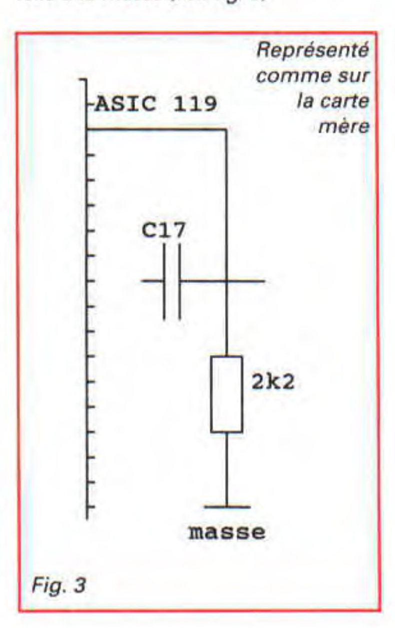


Fig. 3

Étape 5 : maintenant, connectez une résistance 2,2 k/ohms du côté masse de LK4 et une résistance 2,2 k/ohms du côté masse de LK1.
Les autres côtés des résistances seront connectés aux broches 127 et 129 de l'Asic *(voir fig. 4)*.

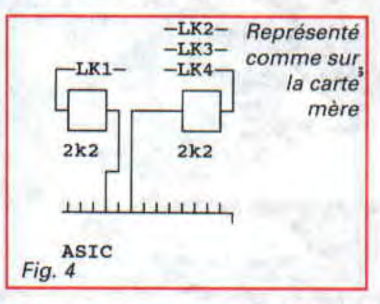


Fig. 4

Étape 6 : coupez la broche 159 de l'Asic sur la carte mère. À la place de l'ancienne connexion, reliez la patte de la dernière résistance 2,2 k/ohms *(voir fig. 5)*.
L'autre côté de la résistance sera connecté avec du 5 V sur la carte mère (les 5 V sont dans le « trou » en face de la broche 159).

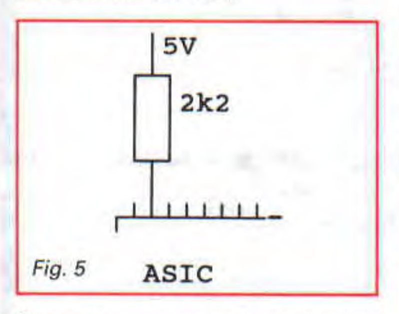


Fig. 5

Étape 7 : connectons le clavier à la GX-4000.
Les broches du clavier se trouvent sur l'Asic de la broche 71 à la broche 80. La broche 80 est la ligne clavier numéro 0, la broche 79 la ligne 1, ..., la broche 71 la ligne 9.
Les lignes du clavier sont sur le circuit sonore (AY-3-8912 ou IC14).
Il est impossible de donner des instructions complètes sur la manière de connecter les broches avec votre clavier CPC, car il existe au moins 6 types de brochages clavier différents utilisés par Amstrad.
La façon la plus simple pour trouver les bonnes broches est de regarder sur votre vieux CPC pour connecter de la même manière les broches du circuit sonore avec le clavier.
Vous trouverez les lignes clavier de votre vieux CPC sur le circuit 74LS145 de la carte mère de votre vieux CPC pour pouvoir faire l'analogie *(voir fig. 6)*.
Connectez-les de la même manière à l'Asic qu'ils le sont sur le 74LS145.

BROCHE	LIGNE CLAVIER
1..7	0..6
9..11	7..9

Fig. 6

L'HEURE DE VÉRITÉ

Étape 8 : votre premier test !
Si tout s'est bien passé jusqu'ici, vous pouvez effectuer votre premier test. Connectez la GX-4000 à votre moniteur (ou téléviseur) et insérez la cartouche CPC+ (et pas la cartouche GX-4000 originale !).
Allumez alors la « console » : sur l'écran devrait apparaître le menu de sélection (F1/F2). Essayez d'accéder au Basic... c'est un premier test pour votre clavier !
Beaucoup d'erreurs peuvent résulter d'une mauvaise connexion du clavier.
Étape 9 : si tout est OK, vous pouvez maintenant vous attaquer au port imprimante (seulement si vous en avez besoin) et au port d'extension.
Je ne détaillerai pas la manière de connecter les broches sur les connecteurs double-faces car vous pouvez trouver le détail des brochages dans tout livre relatif au CPC (comme votre manuel, par exemple).
Port Imprimante : toutes les broches seront trouvées sur l'Asic *(voir fig. 7, 8, 9)*.
Vous pouvez désormais connecter toute extension sur la GX-4000.
Mais si vous voulez connecter votre vieux DDI-1, il y a une petite chose que vous devez corriger. Le DDI-1 contient une Rom interne et la cartouche Rom du CPC+ contient aussi la Rom Amsdos.
C'est pourquoi vous devez « tuer » le système de sélection de votre vieux DDI-1.
Étape 10 : mettre un DDI-1 sur un CPC+ (pas seulement le CPC-GX+).
Ouvrez le contrôleur DDI-1 et retirez la Rom Amsdos. Vous devez alors couper la broche 43 du connecteur sur la carte. Dès lors, votre DDI-1 fonctionnera correctement avec votre CPC+ ou votre GX-4000.
Étape 11 : épongez-vous le front, c'est fini !

MAINTENANT AU BOULOT

Voilà donc la fin de cet article « hardware » qui, je l'espère, vient de vous offrir des sprites hards, 4096 couleurs et pas mal d'autres choses. Je vous incite d'ailleurs à vous procurer le B-Asic qui vous permettra de commencer à utiliser ces nouvelles capacités dans vos œuvres personnelles !
Rappelez-vous aussi que les numéros 38, 39, 40, 41 d'*Amstrad Cent Pour Cent* ont chacun quelques pages spécialement consacrées à la machine.

The Cranium et Longshot

BROCHE	SIGNAL
145	Strobe
146	Data Bit 0
147	Data Bit 1
148	Data Bit 2
149	Data Bit 3
151	Data Bit 4
152	Data Bit 5
153	Data Bit 6
154	Data Bit 7
155	Busy

Fig. 7

Signaux de l'ASIC:

BROCHE	SIGNAL
126	Light Pen
128	Cursor
127	RAMDIS
129	ROMDIS
133	ROMEN
131	RAMRD
159	BUS RESET
119	EXP

Fig. 9

BROCHE	SIGNAL
5	A15
4	A14
3	A13
2	A12
1	A11
40	A10
39	A9
38	A8
37	A7
36	A6
35	A5
34	A4
33	A3
32	A2
31	A1
30	A0
13	D7
10	D6
9	D5
7	D4
8	D3
12	D2
15	D1
14	D0
16	INT
17	NMI
18	HALT
19	MREQ
20	IORQ
28	RFSH
27	M1
26	RESET
25	BUSRQ
24	WAIT (READY)
23	BUSAK
22	WR
21	RD
29	GND
11	5V

Fig. 8

ATTENTION !

La transformation de votre GX4000 et CPC en CPC Plus annule la garantie de votre machine. *Amstrad Cent Pour Cent*, tout comme les auteurs de l'article ne peuvent être responsables de quelques dommages éventuels survenus à votre matériel lors de cette transformation.

EN AVANT L'ASIC

Enfin, les vacances arrivent. Vous allez pouvoir vous consacrer à Cent Pour Cent à votre machine chérie. Alors pour ne pas gâcher votre plaisir, peu de discours ce coup-ci, car j'ai décidé de réserver l'ensemble des pages de cet article à un récapitulatif général des registres de l'Asic.

Le fonctionnement de ces registres a été abordé au cours des différents numéros de votre revue préférée, et la dernière colonne de chaque tableau vous en donne la référence précise.

Qu'est-il possible de rajouter aux connaissances du hardware de l'Asic ? Rappelons tout d'abord la séquence de délockage... en Basic.

```
1000 RESTORE 1010:FOR I=0 TO 16:READ A:OUT &BC00,A:NEXT I:RETURN
1010 DATA 255, 0, 255, 119, 179, 81, 168, 212, 98, 57, 156, 70, 43, 21, 138, 205, 238
```

Lorsqu'une couleur est changée *via* l'ancien système de Gate Array émulé (c'est-à-dire *via* les classiques « out » sur le port #7F00), l'Asic opère une traduction de la couleur passée au format 5 bits (32 couleurs dont 27 réellement différentes) au format 12 bits.

QUELQUES DÉTAILS

Comme nous le voyons dans les tables, les registres de couleurs sont lisibles (Mode Read).

Il est donc aisé de faire la correspondance...

Il suffit de faire un BORDER couleur, par exemple, puis de récupérer le résultat aux adresses #6420 et #6421 de la RamIO Asic.

Quels sont les problèmes d'incompatibilité entre leCPC+ et ses ancêtres ?

Il y a tout d'abord les problèmes de différences entre les Roms. Cela n'étant pas un problème hardware, il a suffit de mettre sur le marché des Roms identiques à celles des anciens modèles... Plus de jeu « Burnin'Rubber », mais aussi plus de risque d'incompatibilité avec vos logiciels adorés tels que Discology.

Viennent ensuite les problèmes hardwares... Les constructeurs de l'Asic ont émulé l'ensemble des circuits de l'ancienne machine tels que le PPI, et c'est sur ce circuit bien précis que les ingénieurs furent trop précis... En effet, le PPI sur les anciens CPC est buggé et la validation (cf. *A100%* n° 43,

Adresses	Taille	Reset	Type	Nom Reg.	Utilisation	Référence A100%
#6434	2	*	R/W	COLS10	Couleur des sprites encre 10	N°38 page 30
#6436	2	*	R/W	COLS11	Couleur des sprites encre 11	"
#6438	2	*	R/W	COLS12	Couleur des sprites encre 12	"
#643A	2	*	R/W	COLS13	Couleur des sprites encre 13	"
#643C	2	*	R/W	COLS14	Couleur des sprites encre 14	"
#643E	2	*	R/W	COLS15	Couleur des sprites encre 15	"
#6800	1		W	PRI	Interruption Raster Programmable	N°47 page 34 à 37
#6801	1		W	SPLT	N° ligne du Split Screen (Rupture)	N°40 page 44 N°47 page 37
#6802	2	*	W	SSA	Adresse écran du Split Screen	N°47 page 37
#6804	1		W	SSCR	Contrôle du retard vidéo	N°45 page 45 N°48 (Ce numéro!)
#6805	1	*	W	IVR	Vecteur Interruption	N°47 page 34 à 36
#6808	1	*	R	ADC0	Entrée analogique canal 0	N°47 page 34 à 37
#6809	1	*	R	ADC1	Entrée analogique canal 1	"
#680A	1	*	R	ADC2	Entrée analogique canal 2	"
#680B	1	*	R	ADC3	Entrée analogique canal 3	"
#680C	1	*	R	ADC4	Entrée analogique canal 4 (Off)	"
#680D	1	*	R	ADC5	Entrée analogique canal 5 (Off)	"
#680E	1	*	R	ADC6	Entrée analogique canal 6 (Off)	"
#680F	1	*	R	ADC7	Entrée analogique canal 7 (Off)	"
#6C00	2	*	W	SAR0	Adresse source canal DMA 0	N°45 page 38 à 40
#6C02	1	*	W	PPR0	Unité de pause canal DMA 0	"
#6C04	2	*	W	SAR1	Adresse source canal DMA 1	"
#6C06	1	*	W	PPR1	Unité de pause canal DMA 1	"
#6C08	2	*	W	SAR2	Adresse source canal DMA 2	"
#6C0A	1	*	W	PPR2	Unité de pause canal DMA 2	"
#6C0F	1		R/W	DCSR	Contrôle/status Dma/Interruption.	N°45 page 38 à 40 N°47 page 34 à 37

page 24) des ports A et C (*via* Bdir et Bc1 à 0 sur le port C) n'est pas nécessaire lors d'une lecture clavier. Quelques programmeurs, pour gagner quelques cycles, se sont donc dispensés de cette validation (fort heureusement, les programmeurs de la Rom n'ont pas fait cette erreur), et c'est pourquoi vous pouvez constater que le clavier ne répond plus sur certains logiciels.

COMMENT FAIRE

La solution au problème ? Il n'y en a qu'une... modifier le logiciel pour y insérer judicieusement l'ordre de validation manquant. Cela nécessite assez souvent quelques jongleries avec le code machine...

Le deuxième cas d'incompatibilité que vous pouvez rencontrer (et qui demeure une bonne méthode de différenciation CPC+/CPC Old) est lié aux interruptions. Si vous vous souvenez du numéro précédent, je vous faisais part de l'état du bit 0 du poids faible de l'adresse du vecteur d'interruption (si, si ! page 34 !). Le mode IM2 étant très rarement utilisé sur les anciens CPC, je pense que seules quelques protections peuvent poser problème. La solution à une galère de ce type reste le décalage de la table de 1 octet ou, mieux encore, l'affectation de la table vectorisée avec des adresses dont le poid faible est identique au poid fort.

Signalons aussi que le CPC+ dispose d'un registre CRTC en lecture à l'adresse #BE00, qui est la copie conforme du registre en #BF00.

Notons, enfin, pour clore le chapitre de la compatibilité, une faiblesse du FDC 765 sur le CPC+ dont certains registres d'état sont faux après une écriture trop « rapide ».

Le moyen le plus sûr pour tester un CPC reste cependant la connexion de sa RamIO Asic.

En effet, Amstrad avait depuis longtemps prévu la sortie du Plus, et les tout derniers modèles de la vieille génération (rarissimes !) sont équipés d'un « pré-Asic », circuit identique à l'Asic dans le sens où il émule le CRTC6845, le PPI8255, le GateArray, le Ay-3-8912, mais ne dispose malheureusement pas des 4096 couleurs (Hé hé, j'en vois certains qui commencaient déjà à rêver...).

J'attends toujours votre courrier concernant le test de l'Asic. Je pense que vous êtes désormais capables de l'écrire. La meilleure solution sera passée dans ces lignes. À vos claviers !

Et... dodo maintenant !

Longshot. Logon System 93

Adresses	Taille	Reset	Type	Nom Reg.	Utilisation	Référence A100%
#6050	2	*	R/W	SPRX10	Position X sprite n°11	N°41 page 51
#6052	2	*	R/W	SPRY10	Position Y sprite n°11	"
#6054	1		W	SPRM10	Zoom du sprite n°11	"
#6058	2	*	R/W	SPRX11	Position X sprite n°12	"
#605A	2	*	R/W	SPRY11	Position Y sprite n°12	"
#605C	1		W	SPRM11	Zoom du sprite n°12	"
#6060	2	*	R/W	SPRX12	Position X sprite n°13	"
#6062	2	*	R/W	SPRY12	Position Y sprite n°13	"
#6064	1		W	SPRM12	Zoom du sprite n°13	"
#6068	2	*	R/W	SPRX13	Position X sprite n°14	"
#606A	2	*	R/W	SPRY13	Position Y sprite n°14	"
#606C	1		W	SPRM13	Zoom du sprite n°14	"
#6070	2	*	R/W	SPRX14	Position X sprite n°15	"
#6072	2	*	R/W	SPRY14	Position Y sprite n°15	"
#6074	1		W	SPRM14	Zoom du sprite n°15	"
#6078	2	*	R/W	SPRX15	Position X sprite n°16	"
#607A	2	*	R/W	SPRY15	Position Y sprite n°16	"
#607C	1		W	SPRM15	Zoom du sprite n°16	"
#6400	2	*	R/W	COLB0	Couleur de fond encre 0	N°38 page 30
#6402	2	*	R/W	COLB1	Couleur de fond encre 1	"
#6404	2	*	R/W	COLB2	Couleur de fond encre 2	"
#6406	2	*	R/W	COLB3	Couleur de fond encre 3	"
#6408	2	*	R/W	COLB4	Couleur de fond encre 4	"
#640A	2	*	R/W	COLB5	Couleur de fond encre 5	"
#640C	2	*	R/W	COLB6	Couleur de fond encre 6	"
#640E	2	*	R/W	COLB7	Couleur de fond encre 7	"
#6410	2	*	R/W	COLB8	Couleur de fond encre 8	"
#6412	2	*	R/W	COLB9	Couleur de fond encre 9	"
#6414	2	*	R/W	COLB10	Couleur de fond encre 10	"
#6416	2	*	R/W	COLB11	Couleur de fond encre 11	"
#6418	2	*	R/W	COLB12	Couleur de fond encre 12	"
#641A	2	*	R/W	COLB13	Couleur de fond encre 13	"
#641C	2	*	R/W	COLB14	Couleur de fond encre 14	"
#641E	2	*	R/W	COLB15	Couleur de fond encre 15	"
#6420	2	*	R/W	COLBORD	Couleur du bord de l'écran	"
#6422	2	*	R/W	COLS1	Couleur des sprites encre 1	"
#6424	2	*	R/W	COLS2	Couleur des sprites encre 2	"
#6426	2	*	R/W	COLS3	Couleur des sprites encre 3	"
#6428	2	*	R/W	COLS4	Couleur des sprites encre 4	"
#642A	2	*	R/W	COLS5	Couleur des sprites encre 5	"
#642C	2	*	R/W	COLS6	Couleur des sprites encre 6	"
#642E	2	*	R/W	COLS7	Couleur des sprites encre 7	"
#6430	2	*	R/W	COLS8	Couleur des sprites encre 8	"
#6432	2	*	R/W	COLS9	Couleur des sprites encre 9	"

Adresses	Taille	Reset	Type	Nom Reg.	Utilisation	Référence A100%
#4000	256	*	R/W	SPRDEF0	Données graphiques sprite n°1	N°41 page 51
#4100	256	*	R/W	SPRDEF1	Données graphiques sprite n°2	"
#4200	256	*	R/W	SPRDEF2	Données graphiques sprite n°3	"
#4300	256	*	R/W	SPRDEF3	Données graphiques sprite n°4	"
#4400	256	*	R/W	SPRDEF4	Données graphiques sprite n°5	"
#4500	256	*	R/W	SPRDEF5	Données graphiques sprite n°6	"
#4600	256	*	R/W	SPRDEF6	Données graphiques sprite n°7	"
#4700	256	*	R/W	SPRDEF7	Données graphiques sprite n°8	"
#4800	256	*	R/W	SPRDEF8	Données graphiques sprite n°9	"
#4900	256	*	R/W	SPRDEF9	Données graphiques sprite n°10	"
#4A00	256	*	R/W	SPRDEF10	Données graphiques sprite n°11	"
#4B00	256	*	R/W	SPRDEF11	Données graphiques sprite n°12	"
#4C00	256	*	R/W	SPRDEF12	Données graphiques sprite n°13	"
#4D00	256	*	R/W	SPRDEF13	Données graphiques sprite n°14	"
#4E00	256	*	R/W	SPRDEF14	Données graphiques sprite n°15	"
#4F00	256	*	R/W	SPRDEF15	Données graphiques sprite n°16	"
#6000	2	*	R/W	SPRX0	Position X sprite n°1	"
#6002	2	*	R/W	SPRY0	Position Y sprite n°1	"
#6004	1		W	SPRM0	Zoom du sprite n°1	"
#6008	2	*	R/W	SPRX1	Position X sprite n°2	"
#600A	2	*	R/W	SPRY1	Position Y sprite n°2	"
#600C	1		W	SPRM1	Zoom du sprite n°2	"
#6010	2	*	R/W	SPRX2	Position X sprite n°3	"
#6012	2	*	R/W	SPRY2	Position Y sprite n°3	"
#6014	1		W	SPRM2	Zoom du sprite n°3	"
#6018	2	*	R/W	SPRX3	Position X sprite n°4	"
#601A	2	*	R/W	SPRY3	Position Y sprite n°4	"
#601C	1		W	SPRM3	Zoom du sprite n°4	"
#6020	2	*	R/W	SPRX4	Position X sprite n°5	"
#6022	2	*	R/W	SPRY4	Position Y sprite n°5	"
#6024	1		W	SPRM4	Zoom du sprite n°5	"
#6028	2	*	R/W	SPRX5	Position X sprite n°6	"
#602A	2	*	R/W	SPRY5	Position Y sprite n°6	"
#602C	1		W	SPRM5	Zoom du sprite n°6	"
#6030	2	*	R/W	SPRX6	Position X sprite n°7	"
#6032	2	*	R/W	SPRY6	Position Y sprite n°7	"
#6034	1		W	SPRM6	Zoom du sprite n°7	"
#6038	2	*	R/W	SPRX7	Position X sprite n°8	"
#603A	2	*	R/W	SPRY7	Position Y sprite n°8	"
#603C	1		W	SPRM7	Zoom du sprite n°8	"
#6040	2	*	R/W	SPRX8	Position X sprite n°9	"
#6042	2	*	R/W	SPRY8	Position Y sprite n°9	"
#6044	1		W	SPRM8	Zoom du sprite n°9	"
#6048	2	*	R/W	SPRX9	Position X sprite n°10	"
#604A	2	*	R/W	SPRY9	Position Y sprite n°10	"
#604C	1		W	SPRM9	Zoom du sprite n°10	"

COLLECTOR (DE T À Z)

N*ous voilà enfin au dernier volet de notre* collector. Vous êtes donc désormais en possession de tous les pokes de la galaxie. Si vous avez manqué le début de nos aventures, n'hésitez pas et commandez les anciens numéros de votre magazine favori à partir du numéro 44.

Vous pensez bien qu'en préparant la fin de ce collector, nous avons mis de côté tous les nouveaux « pokes » que vous nous avez envoyés. Ces derniers seront publiés dès notre prochain numéro.

THE LIGHT CORRIDOR

Voici, pour commencer, les codes de ce jeu :

01 : 0000
02 : 5400
03 : 0101
04 : 3901
05 : 2602
06 : 9902
07 : 4303
08 : 9003
09 : 6904
10 : 3305
11 : 9305
12 : 3406
13 : 0407
14 : 6407
15 : 2008
16 : 7408
17 : 4709
18 : 3810
19 : 0511
20 : 6811
21 : 3212
22 : 0213
23 : 8213
24 : 5014
25 : 1015
26 : 8215
27 : 5116
28 : 0117
29 : 7017
30 : 5518
31 : 2819
32 : 9919
33 : 7320
34 : 2521
35 : 0622
36 : 3722
37 : 1223
38 : 4523
39 : 4124
40 : 1825
41 : 1926
42 : 9726
43 : 5927
44 : 0528
45 : 7328
46 : 3929
47 : 3030
48 : 0531
49 : 8431
50 : 9932

THE STRIDER

Avant de commencer une partie, appuyez simultanément sur les touches Z et 0 (comme zéro). Le bord de l'écran va flasher. Ensuite, vous pouvez commencer votre partie. Pendant celle-ci, appuyez sur la touche H pour mettre le jeu en pause. Ensuite, appuyez simultanément sur les touches Shift et 1 pour avancer dans le jeu. Si vous appuyez sur les touches Shift et 0, alors vous recommencerez le niveau en cours, mais avec toutes vos vies, de plus, le compteur de temps sera réinitialisé.

THE VINDICATOR

Prenez Disco et allez aux rendez-vous suivants. Piste 04, secteur 03, adresse &00B4. Modifiez le 3D en 00.

Cela a pour effet de vous rendre invulnérable pendant le premier niveau.

Piste 17, secteur 41, adresse &11D2. Modifiez le 03 par une valeur de 04 à FF. Cela a pour effet d'augmenter votre nombre de vies pour le deuxième niveau.

Piste 27, secteur 48, adresse &007B. Modifiez le 04 par une valeur de 05 à FF. Vous augmenterez ainsi le nombre de vies pour le troisième niveau.

Enfin, notez que le code pour accéder au niveau 2 est OPPENHEIMER et ENOLAGAY pour le niveau 3.

THEATRE EUROPE

Le code pour lancer les missiles nucléaires est SOLEIL NOIR.

Dans la version en anglais, le code pour les missiles nucléaires est MIDNIGHT SUN

THUNDER BIRDS

Les codes des 3 premiers niveaux sont RECOVERY, ALOYSIUS et ANDERSON.

THUNDER BLADE

En appuyant simultanément sur les touches Control, Caps-lock et U, S, G et O avant de commencer le jeu, vous pourrez, par la suite, changer de niveau à n'importe quel moment en tapant simplement sur la touche Return. Cool non ?

THUNDER BIRDS

Les codes pour accéder directement aux diverses parties du jeu sont :

2ᵉ niveau : RECOVERY
3ᵉ niveau : ALOYSIUS
4ᵉ niveau : ANDERSON

THUNDER CATS

Voici la méthode pour être invulnérable dans ce gigantesque jeu. Recherchez tout simplement la chaîne hexa C0,DD,7E,00,87 et remplacez le C0 par un C9. À propos, écoutez la musique de ce jeu, elle est merveilleuse.

TIME SCANNER

Recherchez avec Disco la chaîne hexa 3E,00,32,BD,00,3E et remplacez le premier zéro par le chiffre du niveau.

Mieux encore, avec la multiface, pokez : POKE &1021, le numéro de tableau de départ (de 0 à 3).

TITAN

Le fabuleux jeu de Philippe Pamart vous donne rendez-vous piste 20, secteur 00, adresse &0541, pour modifier le 57,C5 par 2A,C7. L'effet ? Cela permet de passer au tableau suivant en appuyant sur Escape.

Pour la version « Les 100 % A d'OR » et avec la multiface : POKE &1225,&FF vous donne un max de vies supplémentaires.

TORTUES NINJA II

Il s'agit d'un cheat mode : redéfinissez le « Player 2 » au clavier en tapant CHEAT. Ensuite, vous pouvez changer de niveau en tapant P puis la barre espace. Vous avez aussi des vies infinies. Avec ça, si vous n'arrivez pas à la fin, eh bien ! je ne peux plus rien pour vous.

Si vous avez Disco sous la main, allez en piste 18, secteur 48 ensuite recherchez la chaîne 28,04,36,00,18 et remplacez le 36 par un 39 pour avoir les vies infinies pour les 2 joueurs.

RÉSUMÉ:
POKES S'INTERROGE:
QUE VA DONC FAIRE POUM AVEC SON TABOURET ET SA CORDE ?...
QUAND LES TEMPS SONT

LES JEUX SONT FAITS

DURS, QUE PEUX-T-ON FAIRE D'UN TABOURET ET D'UNE CORDE ?...
POKES S'INTERROGE...
UNE CORDE... UN TABOURET...

















HÉHÉ... IL Y A DE NOUVEAUX JEUX? L'ÉTÉ SERA CHAUD! LE MACRAMÉ SURVIVRA-T-IL ?... POUM SE REMETTRA-T-IL DE SA CHUTE ?... LES SÉQUELLES NE SERONT-ELLES PAS IRRÉVERSIBLES ?... LE DIAGNOSTIC DANS LES PROCHAINS NUMÉROS!...

MYKAÏA

TOTAL RECALL
Pour les multifaceurs, POKE &0B36,&0F : élimination des adversaires avec une seule balle.
POKE &0B3B,&0F : 95 % d'énergie.
POKE &8600,&01 : en appuyant sur Return vous activez le « test mode » (passage niveau suivant).
Pour les Discomen, allez en piste 13, secteur 42, à l'adresse &00FC où vous placerez un 00. Puis, piste 24, secteur 47, à l'adresse &00EF où vous placerez un 01 pour passer les niveaux en appuyant sur la touche Return.
Un p'tit cheat ? Pour changer de level dans le tableau des scores, tapez « THE END IS NIGHT ».

TRAILBLAZER
Voici une bidouille pour avoir 255 sauts pour ce soft qui est considéré comme l'un des meilleurs jeux jamais créés sur CPC par les Logon System, c'est dire la qualité du soft. Recherchez la chaîne 83-26-3E-04-32-BA et remplacez le 04 par un joli petit FF.

TRANTOR
Allez Disco ! Recherchez la chaîne : 79,00,5A,00,00 et remplacez cette dernière par : 79,00,FF,00,00. Cela aura pour effet de vous donner 255 secondes par niveau.
Les codes du jeu sont :
KEMPSTON
JOYSTICK
SPECTRUM
SOFTWARE
KEYBOARD
COMPUTER
CASSETTE
SINCLAIR
GRAPHICS
HARDWARE
TERMINAL
Allez, on continue, recherchez 3A,18,0C,3D,32,18,0C,3A et remplacez le 3D par 00 pour obtenir du fuel infini pour votre lance-flammes.
Maintenat, pour finir recherchez 3A,5B,0C,3D,32,5B,0C,DD et remplacez le 3D par 00. Ne me remerciez pas, c'est normal.

TURRICAN
En cours de partie, baissez-vous et appuyer sur Shift et Return pour être invulnérable.
Avec la Multiface, POKE &04F0,&00 pour de l'énergie infinie.
Autre cheat. Au début du jeu, mettez la pause, puis appuyez simultanément sur les touches RAIMBOW. Vous obtiendrez 99 vies et aussi le nombre maximum d'énergie pour vos armes spéciales. (valable pour les claviers AZERTY).

TURRICAN 2
Pour transformer le « Turrican » en boule et le rendre invincible, appuyez en diagonale et vers le bas sur le joystick, et appuyez en même temps sur la touche Return.

TURTLES
Faites une recherche avec Disco en piste 21, secteur C6, zones 0120 et 0140. Vous trouverez des 3D,32,7E,86,06,7F,0E,10 et vous remplacerez les 3D par des 00 pour avoir des vies infinies.

TURTLES 2
Avec la Multiface :
POKE &15DC,&00 pour des crédits infinis.
POKE &1667,&00 pour l'invulnérabilité.

TWINS
Pour avoir 255 vies, tapez dans le lanceur après avoir chargé twins1.bin et twins2.bin, POKE 30283,255:CALL 30000
De même pour de l'énergie infinie, ajoutez la ligne 111 : POKE &8027,&C9.

UN SQUADRON
Redéfinissez les touches de déplacements. Pour cela, il faut prendre le joystick 1 pour les 2 joueurs, puis sélectionner le jeu à 2 joueurs et ensuite tirer sur le manche. Les 2 avions se rejoindront et quand vous tirerez, le tir se décuplera. Sympa comme truc !

VICTORY ROAD
Toujours avec l'ami Disco, recherchez la chaîne suivante :
22,57,13,3E,06,32,FD,CE
et remplacez cette dernière par :
22,57,13,3E,FF,32,FD,CE.
Cela vous donnera 255 vies.

VIGILANTE
Cherchez la chaîne :
C2,BD,4C,18,E2,3E,03,32,01,FE
Remplacez-la par :
C2,BD,4C,18,E2,3E,FF,32,01,FE
Vous voilà avec 255 vies.
Mieux encore, avec la Multiface :
POKE &8C2B,&A7 pour des vies infinies.
Si le cœur vous en dit, toujours avec la Multiface, pokez :
POKE 5266,&FF pour un max' de vies.

WEC LE MANS
Pour obtenir du temps à l'infini dans ce jeu, ce qui est bien pratique pour être sûr de le terminer un jour, il suffit de rechercher la chaîne hexa D6,01,27,77,23 et de remplacer le 01 par un 00.
Enfin, pour obtenir un véritable anticrash, recherchez la chaîne hexa 08,3D,32,D0,08,01 et remplacez le 3D par un 00.
Vous pourrez ainsi vous permettre de jouer les autos tamponneuses avec les voitures adverses ou de sortir largement d'un virage sans craindre quoi que ce soit des panneaux publicitaires Ocean. À chaque crash, vous rétrograderez simplement en première vitesse, mais ça, c'est pas trop méchant.
Pour ce jeu avec la Multiface :
POKE &1345,&00 vous donne du temps infini.
POKE &0C5F,&00 vous donne l'anticrash.
Je sais, il y a aussi les trucs avec Disco. Recherchez donc la chaîne hexa D6,01,27,77,23 et remplacez le 01 par un 00 pour avoir du temps infini.
De même, recherchez la chaîne hexa 08,3D,32,D0,08,01 et remplacez le 3D par un 00 pour l'anticrash.

WEST PHASER
Attention ! cette bidouille ne fonctionne qu'avec la première version du soft (celle fournie avec le phaser de Loriciel). Alors, pour un nombre de balles illimité, il faut modifier sur la face B de la disquette le secteur C5 de la piste 20, et modifiez le 3D trouvé par 00.

WIND SURF WILLY
Avec votre Multiface favorite, pokez &15E6,&00 et vous voilà avec du temps infini.

WING OF FURY
Vous voulez des vies infinies ? Pas de problème. En piste 10, secteur 43, à l'adresse #00A0, vous trouverez sans problème la chaîne 21,8F,00,7E,A7,20,33,3A. Il suffit de remplacer le #20 par #18.

WONDER BOY
Allez, un p'tit rendez-vous de plus. Piste 13, secteur 00, adresse &02A5. Modifiez le 35 par 00. Effet : vies infinies.

XENON II
Hugh ! Pour pouvoir allumer facilement le vilain-pas-beau de service, il faut faire RUN"PAIS" au lieu de RUN"DISC". Ah ! j'oubliais, c'est sur la compil' Edition One.

XYPHOES FANTASY
Recherchez la chaîne hexa : 80,03,08,08,04,00,00,00 et remplacez le dernier 00 par un 01 pour avoir des vies infinies.
Avec la Multiface :
POKE &12,&01 pour les mêmes vies infinies.
Le cheat mode de Xyphoes s'active pendant le menu. Il faut appuyer dans le pavé numérique sur les touches suivantes : F9, F6, F3, F2, F5. Il faut aussi appuyer sur le « point ». Cela donnera des vies infinies et de l'énergie infinie pour les combats.

YIE AR KUNG FU
Recherchez avec Disco la chaîne 67,87,3D,32 et remplacez par 67,87,3D,35 pour être invincible.
Dans « Collection Ocean » recherchez la chaîne 3D,32,FA,88,21 et remplacez-la par 00,32,FA,88,21.
Ouf ! Ce fut long mais bon. Je vous conseille vivement de collectionner ces pokes, car tôt ou tard, un ami voire vous-même se trouvera bloqué dans un jeu. Quel pied alors de trouver le cheat ou le poke qui vous dépannera.

Poke &XX,243

La Classe DES LECTEURS

Ainsi que la tradition de *Cent Pour Cent* l'exige, dans ce numéro, un de nos charmants lecteurs va pouvoir commander 750 F de matériel informatique chez Jessico. Un deuxième gagnant se verra attribuer un bon de commande de 500 F chez le même revendeur.

Cette fois-ci, il s'agit de Yves Gerey (gagnant disquette) et de Bernard Marschal (gagnant Minitel). Je vous sens tous un peu jaloux. Alors, si vous voulez faire comme eux, vous pouvez procéder de deux manières : envoyer vos écrans sur disquette au 19, rue Louis-Pasteur, 92513 Boulogne Cedex, ou transférer par le biais du Minitel vos plus belles images. En tout cas, ne quittez pas vos écrans, et à la prochaine pour une autre sélection...



Notre gagnant Minitel : Bernard Marschal.

Cela pourrait être l'adaptation de « The 7th Guest » sur CPC, Bravo Wilfrid Tourbier.

Superbe rhinocéros en mode 2, 3 couleurs, overscan. Yves Gerey en est le créateur.

Enfin un logo à la hauteur du journal : merci Ludovic Wahnert.



Sébastien Bernard et sa supergrenouille.



LE SERVEUR EN FOLIE

Comme à l'accoutumée, nous revoilà pour une sélection de nos meilleurs jeux téléchargeables sur CPC. En effet, n'oubliez pas que par le biais d'une machine formidable vous avez accès à une logithèque de plus de 300 softs. Alors... alors... heureux ?

Je vous rappelle brièvement comment cette noble magie nommée « téléchargment » opère dans l'esprit tordu de ces fanas de CPC. En connectant votre cher Amstrad au Minitel par un câble fantastique, vous pouvez transférer des centaines de jeux sur vos disquettes pour un coût dérisoire ! D'autre part, sur ce serveur merveilleux, vous pourrez également participer à des concours, gagner des jeux, dialoguer, échanger des fichers, vendre vos originaux, lancer des SOS... Enfin bref, le 3615 ACPC est le lieu de rencontre pour tous les mordus. Alors rejoignez vite le club. Tapotez sur votre Minitel 3615 ACPC, et que la Force soit avec vous !

Allez, allez, je vais vous présenter quelques-uns des chefs-d'oeuvre disponibles sur notre serveur.



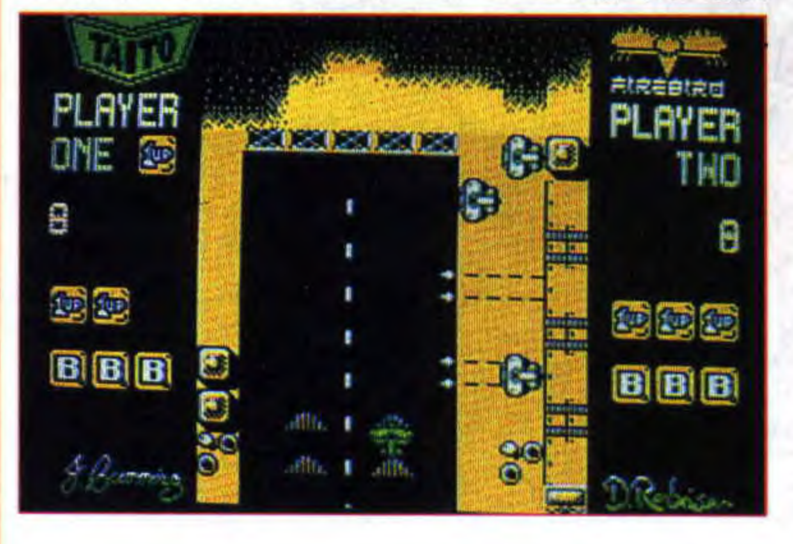
ARKANOID 2

Eh oui, voici un des plus beaux jeux qu'ait connu le CPC. Ce soft, qui rencontra un vaste succès planétaire est désormais accessible à tous pour douze misérables minutes de 3615. Vous pourrez donc continuer sur les traces d'Arkanoid, et peut-être affronter le terrible Dho. Mais avant cet ultime combat sans merci, il vous faudra des nerfs d'acier pour passer tous les niveaux de ce casse-briques, ô combien fabuleux, grâce aux nombreux bonus que vous pourrez rafler. Alors, préparez aspirine et patience, Arkanoid 2 vous procurera le reste : la joie, la bonne humeur, et le plaisir d'un jeu bien fait.

CAULDRON 2

À l'heure où Cauldron 3 vient de sortir, il est bon de compléter sa logithèque avec un soft digne des quelque dix-huit minutes de 3615 qu'il vaut largement. En effet, vous incarnerez la fameuse citrouille devant détrôner la méchante-pas-belle-vilaine-pas-gentille sorcière qui fait régner sur le Royaume un chaos absolu. Pour accomplir votre quête, vous serez seul, avec comme unique arme votre magie. Y arriverez-vous ? Serez-vous assez agile pour surmonter tous les obstacles ? Survivrez-vous aux pièges démoniaques ? Et, enfin, détruirez-vous l'infecte sorcière ? Autant de questions passionnantes qui ne trouveront de réponses que dans le sein d'un serveur connu de tous : ACPC. Que la magie soit avec vous ! *Amen.*





CRAZY BALL

L'histoire est tout simplement formidable. Imaginez un peu un clown nommé Boubou, qui fait rire les enfants, son succès est international. Au moment de son apogée, arrive le clown Duglan qui lui dérobe subreptícement son nez rouge ! Horreur, la déchéance ne va pas tarder s'il ne retrouve pas son nez. Votre mission : vous êtes le nez et vous devez parcourir un labyrinthe complexe afin de retrouver votre patron et faire enfin revivre le sourire sur les visages de ces charmantes têtes blondes (et brunes, et rousses). Donc Crazy Ball, est un jeu tout mignon tout sympa qui vous fera passer de



bonnes et longues heures devant votre CPC. Avouez, tout de même, que vous êtes vraiment gâtés ! Alors, maintenant que vous connaissez l'essentiel du jeu, la balle est dans votre camp!

GHOST-BUSTERS

Les revoilà ! Les fameux chasseurs de fantômes sont de retour sur notre machine. Ainsi, vous pourrez incarner tour à tour Peter, le professeur fou ou l'un des deux autres hurluberlus qui constituent cette équipe formidable. Eddy, par exemple, s'est adonné à ce jeu pendant de nombreuses semaines, il n'en n'est pas encore revenu. Alors, bouffé par un fantôme de la moquette caprie ? Ionisé par ses armes fulgurantes ? Envoyé dans un monde parallèle (d'où il venait) ? Ou tout simplement s'éclate-t-il encore sur ce soft ? La réponse est là, à quelques mètres de moi, où une étrange musique règne : la, la, la, la, la, la... Les Ghostbusters auraient-ils eu raison d'Eddy ? La suite au prochain numéro...

FLYING SHARK

À tous les passionnés de Gronavion, à tous ceux que le manche à balai laisse rêveurs, à ceux qui ne supportent pas de rester trop longtemps sur le plancher des vaches, à tous ceux





qui veulent connaître de nouvelles sensations, à tous ceux qui sont allés sur Mars, à tous ceux qui se sont lamentablement crashés à plus de 8 000 mètres d'altitude, enfin bref, à tout le monde : essayez Flying Shark, un jeu certes un peu faiblard en couleurs, mais passionnant. Imaginez donc un peu un combat aérien à bord d'un vieux coucou de la seconde guerre mondiale, imaginez des centaines d'avions se ruant sur vous dans un seul but : vous anéantir. Ça y est, vous y êtes. Vous voyez Flying Shark. Joli, non ? Mais attention aux effets secondaires ! Une fois qu'on a commencé à y jouer, on ne peut plus s'en passer...

NEBULUS

Le revoilà notre petit bonhomme vert, ou plutôt devrais-je dire notre petit extraterrestre vert. Vous devrez le guider dans un labyrinthe compliqué comme il se doit, et où vos nerfs seront encore une fois mis à rude épreuve. Pardi, ce labyrinthe est en fait une tourelle qui tourne sur elle-même, et, en franchissant les portes ou en montant les escaliers, vous pourrez progresser. Mais attention ! plus vous serez haut et plus dure sera la chute, alors prudence... Enfin bref, ce hit connu de tous doit se trouver dans votre logithèque afin de ne pas avoir l'air bête. Alors foncez sur votre Minitel, 3615 ACPC rubrique 5.

LE MOT DE LA FIN

Eh oui, c'est déjà fini. Je sais, c'est trop court, mais attendez un peu le prochain numéro, et vous allez voir ce que vous allez voir : une nouvelle sélection de nos meilleurs jeux ! Maintenant, je vous laisse en compagnie du Minitel, de la ravissante, charmante et sublimissime Barbie, la généreuse maîtresse des lieux, qui ne manquera pas de vous faire gagner de superbes lots. Sur ce, bon Minitel, bon 3615, bon téléchargement...

Totov, le téléchargé du Minitel de la mort qui tue

Retrouvez la liste complète des jeux à télécharger sur 3615 ACPC

AVANT-GOÛT DE VACANCES...

Malgré les quelques turbulences que traverse le CPC en ce moment, il est agréable de constater, non sans un peu de surprise et de satisfaction, que la production des fanzines reste invariable depuis plus d'un an !

Ainsi donc, les « rédac'chefs fanzines » demeurent attachés à leur bonne vieille machine qu'est l'Amstrad. À tous ceux-là, je tiens à dire un grand BRAVO ! Bravo messieurs, c'est en partie grâce à vous que le CPC survit, et grâce à vous que nos cellules grises s'excitent sur de nouveaux écrans, de nouveaux programmes... Il semble évident que nos vacances d'août seront largement agrémentées de nombreux fanzines. Pour orienter votre choix, ou tout simplement découvrir de nouveaux chefs-d'œuvre : suivez le guide ....

MICRO MAG & PRESS FIRE 7

Avant toute chose, il me semble important de préciser que *Micro Mag* fait certainement partie des précurseurs des « european fanz ». En effet, toute l'Europe pourra à présent lire ce fanzine : les articles français sont traduits (des fois sommairement, mais tout de même) en anglais : chapeau les mecs, ça c'est du boulot ! Commençons par le premier menu, eh bien ! sachez que le programme se charge de reconnaître votre type de CRTC afin de pouvoir afficher le plus proprement possible les rasters, les images scrollantes, la fenêtre redimensionnée, enfin en un mot comme en mille, les concepteurs ont tout prévu : les pages menu sont vraiment belles.



Maintenant parlons un peu du contenu qui représente environ 250 Ko de texte ! Pas mal, hein ! Essayons d'en faire le tour sommairement. Vous pourrez trouver sur la première face des tests de softs sur CPC, tels que Lemmings (47 %), Atomic (72 %), Super Cauldron (87 %), Soundtrakker, The Tiny Skweeks (48 %). Vous pourrez également prendre des nouvelles de Face Hugger et de Tom & Jerry puisqu'ils sont tous deux interviewés. La traditionnelle rubrique Assembleur est également fidèle au poste. Vous pourrez apprendre les tables de codes, la division... Vous pourrez également trouver 17 tests de fanzines. Une étrange rubrique « Message fore the masses » vous intriguera sûrement. Eh oui, il s'agit d'une rubrique Poke qui offre 6 séries de codes. Passons tout de suite à la deuxième face où tous les petits potins du CPC vous seront contés dans la rubrique News. Vous pourrez également trouver 3 tests de démos. Mais chose importante, si comme moi vous aimez bricoler votre CPC, si vous n'avez pas peur d'y introduire bout de câbles et fer à souder, ne ratez pas la troisième rubrique qui détaille le Reset et la Pause hard ! Je n'ai plus assez de place pour continuer, mais il faut absolument savoir que si ce fanzine devait être comparé, ce serait à Arkadia !

Micro Mag & Press Fire 7
compatible CPC+
Nicolas Miraucourt
34, rue du Donjon
77310 Boissise-le-Roi

DARKSTRAD N° 2

Avis à tous les mordus de *Terminator 2*, vous ne pouvez pas manquer Darkstrad. En effet, les nombreuses digits de Schwarzy donnent un aspect musclé à ce fanzine déjà costaud ! Bon, je ne peux résister à l'envie de vous en dévoiler un peu plus. Dès la première face, vous trouverez l'édito, la rubrique Top News (les news du CPC), et toute une initiation à la programmation des Outs Basic. Sur la face B, vous aurez la joie de retrouver les tests de Zap't'Balls (qui obtient de bonnes notes), Sprites Alive (15/20), Prehistorik 2 (15/20), The Addams Family (92 %). Une toute petite question : « Pourquoi mettez-vous aléatoirement des notes sur vingt et sur cent ? ». Il s'agit là d'un détail, j'en conviens, mais quand même... Toujours sur la même face, vous trouverez des programmes, des trucs, enfin bref, toutes ces petites choses qui rendent la vie tellement plus facile aux programmeurs ! Vous pourrez également vous initier à la programmation de l'Asic sur CPC Plus, et accrochez-vous : une démo vous attend. Oui oui, vous avez bien entendu : une démo


THE BIG BOSS NO. 3

SOMMAIRE

EDITORIAL

sur CPC Plus ! J'aimerais encore vous parler de la deuxième disquette (en effet, ce fanzine tient sur deux disquettes archi pleines). M'enfin, tant pis, sachez qu'il vous faut ce fanzine, et c'est tout...

Darkstrad n° 2 CPC 6128 et Plus
Yann Espenel
Les-Bouchets 07460 Beaulieu

THE BIG BOSS N° 3

The *Big Boss* est un fanzine papier (comme vous pouvez le voir sur les photos d'ailleurs) de 17 pages. Mais il ne s'agit pas ici de 17 pages « petit-budget-maquette-bâclée », il s'agit bien de 17 pages aux textes condensés, propres, avec une maquette très réussie... Ainsi la première page (après le sommaire, il va de soi) est occupée par la rubrique Fanzine, vous y trouverez *Le Canard déchaîné*, *Le Petit Electro-Jack illustré*, *Quasar*, *Disc Full*, *Pot de Call*, enfin bref, rien que des grands noms du CPC. La deuxième page est consacrée aux démos (Rourke Demo, surtout), mais je ne m'étendrai pas plus sur le sujet... Car arrivent les tests, ah ah, apparemment, cela vous intéresse ! Eh bien, ces tests concernent Italy 1990 (sur CPC, 62,2 %), Addams Family 2 (sur Super Nintendo, 97 %), Les Tiny Toon's (sur Megadrive, 96 %). Maintenant, si l'électronique vous passionne, et si le cœur vous en dit, vous irez faire un petit tour du côté de la page 6 où vous apprendrez comment faire une animation lumineuse sur des diodes électro-luminescentes.

Quelques pages après, une petite surprise vous attend : non pas un, ni (chon) deux, mais trois listings ! Chose qui devient de plus en plus rare dans nos bons vieux fanzines. Juste après, des sondages, l'interview de CJC, des pokes, des bidouilles, des cheat mode, encore des programmes. Ensuite arrive la rubrique 3D Construction Kit, qui mérite que l'on s'y attarde quelque peu, car grâce à cette rubrique, vous apprendrez la programmation sous ce sublime logiciel. Enfin, le fanzine s'achève sur le traditionnel questionnaire dépouilllé dans le numéro suivant.

Bruno Moreno
2, allée des Lilas
45210 Fontenay/s/Loing

LE MINITEL : DU 100% FANZINES

Eh oui, il y a du changement du côté du Minitel ! Plusieurs mesures ont été prises. Tout d'abord, l'ancienne liste des adresses des différents fanzines testés a été réactualisée. Mais surtout, il est désormais possible de rentrer son fanzine dans cette liste le plus



simplement du monde. Pour cela, allez à la rubrique Fanzine, tapez sur la touche Guide et entrez les différentes informations qui vous sont demandées. Et voilà, le tour est joué ! C'est pas beau la vie ? Ces vacances vont être démentes, je le sens...

LES NOUVELLES DU FRONT

Viennent de paraître : *Le Canard amoché* n° 8, *Croco Passion* n° 9, *Electro Jack* n° 8, *Maxi Micro* n° 9, *Disc Full* n° 9, *Croco Computer Club* n° 7, *Info System CPC* n° 5, *Help Fanz*, *CPC+ Fanz*, *Perfect Fanz*, *Test Fanz*, *Demo Fanz*.

Ont arrêté : *Reset* (possibilité de commander les anciens numéros), *Croco Passion (idem)*.

Pixel n° 2 est prévu pour janvier 1994.

Deux rendez-vous à ne pas manquer pendant vos vacances, celui du 23 au 26 juillet (inclus) à Ferrières-en-Gâtinais (renseignements : Bruno Moreno, 2, allée des Lilas 45210 Fontenay/s/Loing) et celui du week-end du 28-29 août, à Masevaux (renseignements : Jérôme Scham, 8, passage du Cdt-Berger, 68290 Masevaux).



Qu'ils sont beaux tous ces fanzines ! Que vous allez passer de bons moments pendant toutes ces vacances ! Oui, mais comment commander ces fanzines ? C'est plus que simple. Il suffit d'écrire à l'adresse notée en bleu à la fin du test, en envoyant le nombre de disquettes correspondant, ainsi qu'une enveloppe auto-adressée à 4,80 F. Et les charmants rédac' chefs se feront un plaisir de vous communiquer leur passion : le CPC.

Pour toutes les nouvelles sur les fanzines, le Minitel reste à votre disposition : BAL Totov.